

UNIVERSIDAD NACIONAL DE PIURA
FACULTAD DE INGENIERÍA INDUSTRIAL
ESCUELA PROFESIONAL DE INGENIERÍA INFORMÁTICA

**Programa de Actualización para la Obtención de Título
Profesional – PATPRO XIX**



INFORME DE INVESTIGACIÓN

**“PROTOTIPO DE SISTEMA DE MONITOREO DE ALARMAS
EN UNA RED DE DATA CENTERS”**

PARA OPTAR EL TÍTULO PROFESIONAL DE INGENIERO INFORMÁTICO

Presentado por:

Br. José Antonio Burneo Chávez

Br. Kenmy Isrrael Jacobo Montes Torres

Br. Pedro David Saavedra Puchulán

Línea de Investigación: Informática, Electrónica y Telecomunicaciones.

Sub línea de Investigación: Plataforma de TIC.

Piura - Perú

2019

UNIVERSIDAD NACIONAL DE PIURA
FACULTAD DE INGENIERÍA INDUSTRIAL
ESCUELA PROFESIONAL DE INGENIERÍA INFORMÁTICA

**Programa de Actualización para la Obtención de Título
Profesional – PATPRO XIX**



INFORME DE INVESTIGACIÓN

**“PROTOTIPO DE SISTEMA DE MONITOREO DE ALARMAS
EN UNA RED DE DATA CENTERS”**

**LOS SUSCRITOS DECLARAMOS QUE EL PRESENTE INFORME DE
INVESTIGACIÓN ES ORIGINAL, EN SU CONTENIDO Y FORMA.**

Br. José Antonio Burneo Chávez
Ejecutor

Br. Kenmy Israel Jacobo Montes Torres
Ejecutor

Br. Pedro David Saavedra Puchulán
Ejecutor

Ing. Hebert Eduardo Espino Aguirre
Asesor

DECLARACIÓN JURADA DE ORIGINALIDAD DEL INFORME DE INVESTIGACIÓN

Yo: **JOSÉ ANTONIO BURNEO CHÁVEZ** identificado con DNI N° 71538091,
Bachiller de la Escuela Profesional de **Ingeniería Informática** de la Facultad de
Ingeniería Industrial y domiciliado en Calle Bolívar N° 694, del Distrito de Sullana,
Provincia de Sullana, Departamento de Piura, Celular 976870837, Email:
josea.burneo@gmail.com

**"PROTOTIPO DE SISTEMA DE MONITOREO DE ALARMAS EN UNA RED DE
DATA CENTERS"**

DECLARO BAJO JURAMENTO: que el Informe de Investigación que presento
es original e inédito, no siendo copia parcial ni total de una tesis desarrollada, y/o
realizada en el Perú o en el Extranjero, en caso contrario de resultar falsa la información
que proporciono, me sujeto a los alcances de los establecido en el Art. N° 411, del código
Penal concordante con el Art. 32° de la Ley N° 27444, y Ley de Procedimientos
Administrativo General y las Normales Legales de Protección a los Derechos de Autor.

En fe de lo que digo, firmo la presente.

Piura, 19 de julio del 2019.



Br. José Antonio Burneo Chávez
DNI N° 71538091



Artículo 411.- El que, en un procedimiento administrativo, hace una falsa declaración en
relación a hechos o circunstancias que le corresponde probar, violando la presunción de
veracidad establecida por la ley, será reprimido con pena privativa de la libertad no menor
a uno ni mayor de cuatro años.

**Artículo 4, Inciso 4.12 del Reglamento del Registro Nacional de Trabajos de Investigación
para optar grados académicos y títulos profesionales – RENATI Resolución de Consejo
Directivo N° 033-2016-SUNEDU/CD**

DECLARACIÓN JURADA DE ORIGINALIDAD DEL INFORME DE INVESTIGACIÓN

Yo: **KENMY ISRRAEL JACOBO MONTES TORRES** identificado con DNI N° **47005055**, Bachiller de la Escuela Profesional de **Ingeniería Informática** de la Facultad de **Ingeniería Industrial** y domiciliado en Jr. Nicolás de Piérola N° 434 Urb. Ingeniería, del Distrito de San Martín de Porres, Provincia de Lima, Departamento de Lima, Celular: 957548345, Email: kmontestx@gmail.com.

"PROTOTIPO DE SISTEMA DE MONITOREO DE ALARMAS EN UNA RED DE DATA CENTERS"

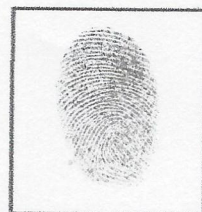
DECLARO BAJO JURAMENTO: que el Informe de Investigación que presento es original e inédito, no siendo copia parcial ni total de una tesis desarrollada, y/o realizada en el Perú o en el Extranjero, en caso contrario de resultar falsa la información que proporciono, me sujeto a los alcances de lo establecido en el Art. N° 411, del código Penal concordante con el Art. 32° de la Ley N° 27444, y Ley de Procedimientos Administrativo General y las Normales Legales de Protección a los Derechos de Autor.

En fe de lo que digo, firmo la presente.

Piura, 19 de julio del 2019.



Br. Kenmy Israel Jacobo Montes Torres
DNI N° 47005055



Artículo 411.- El que, en un procedimiento administrativo, hace una falsa declaración en relación a hechos o circunstancias que le corresponde probar, violando la presunción de veracidad establecida por la ley, será reprimido con pena privativa de la libertad no menor a uno ni mayor de cuatro años.

Artículo 4, Inciso 4.12 del Reglamento del Registro Nacional de Trabajos de Investigación para optar grados académicos y títulos profesionales – RENATI Resolución de Consejo Directivo N° 033-2016-SUNEDU/CD

DECLARACIÓN JURADA DE ORIGINALIDAD DEL INFORME DE INVESTIGACIÓN

Yo: **PEDRO DAVID SAAVEDRA PUCHULÁN** identificado con DNI N°
70612145, Bachiller de la Escuela Profesional de Ingeniería Informática de la Facultad
de Ingeniería Industrial y domiciliado en Jirón Porvenir N° 385, del Distrito de Sullana,
Provincia de Sullana, Departamento de Piura, Celular: 971734374, Email:
pdsaavedrap@gmail.com

"PROTOTIPO DE SISTEMA DE MONITOREO DE ALARMAS EN UNA RED DE
DATA CENTERS"

DECLARO BAJO JURAMENTO: que el Informe de Investigación que presento
es original e inédito, no siendo copia parcial ni total de una tesis desarrollada, y/o
realizada en el Perú o en el Extranjero, en caso contrario de resultar falsa la información
que proporciono, me sujeto a los alcances de los establecido en el Art. N° 411, del código
Penal concordante con el Art. 32° de la Ley N° 27444, y Ley de Procedimientos
Administrativo General y las Normales Legales de Protección a los Derechos de Autor.

En fe de lo que digo, firmo la presente.

Piura, 19 de julio del 2019.

PDS

Br. Pedro David Saavedra Puchulán
DNI N° 70612145



Artículo 411.- El que, en un procedimiento administrativo, hace una falsa declaración en
relación a hechos o circunstancias que le corresponde probar, violando la presunción de
veracidad establecida por la ley, será reprimido con pena privativa de la libertad no menor
a uno ni mayor de cuatro años.

**Artículo 4, Inciso 4.12 del Reglamento del Registro Nacional de Trabajos de Investigación
para optar grados académicos y títulos profesionales – RENATI Resolución de Consejo
Directivo N° 033-2016-SUNEDU/CD**



UNIVERSIDAD NACIONAL DE PIURA
PROGRAMA DE ACTUALIZACIÓN PROFESIONAL
PATPRO EN INGENIERÍA INFORMÁTICA
VERSIÓN XIX - 2019



**ACTA DE EVALUACIÓN DEL INFORME DE
INVESTIGACIÓN**

Los Miembros del Jurado Calificador del Informe de Investigación denominado **"PROTOTIPO DE SISTEMA DE MONITOREO DE ALARMAS EN UNA RED DE DATA CENTERS"** presentado por los bachilleres: **BURNEO CHAVEZ JOSE ANTONIO, MONTES TORRES KENMY ISRRAEL JACOBO Y SAAVEDRA PUCHULAN PEDRO DAVID** participantes del Programa de actualización para Titulación Profesional en la Especialidad de Ingeniería Informática Versión XIX 2019; asesorados por el Mg. Hebert Espino Aguirre. Revisado y absueltas las observaciones formuladas por el Jurado Calificador, lo declaran:



Con la nota:

BURNEO CHAVEZ JOSE ANTONIO

14

MONTES TORRES KENMY ISRRAEL JACOBO

14

SAAVEDRA PUCHULAN PEDRO DAVID

14

Piura, 13 de julio del 2019

.....
Mg. **JORGE SANDOVAL RIVERA**
Miembro del Jurado

.....
Dr. **RIGO REQUENA FLORES**
Miembro del Jurado

.....
Mg. **JUAN MANUEL JACINTO SANDOVAL**
Miembro del Jurado

UNIVERSIDAD NACIONAL DE PIURA
FACULTAD DE INGENIERÍA INDUSTRIAL
ESCUELA PROFESIONAL DE INGENIERÍA INFORMÁTICA

**Programa de Actualización para la Obtención de Título
Profesional – PATPRO XIX**



INFORME DE INVESTIGACIÓN

**“PROTOTIPO DE SISTEMA DE MONITOREO DE ALARMAS EN
UNA RED DE DATA CENTERS”**

APROBADA EN CONTENIDO Y ESTILO POR:

Dr. Rigo Félix Requena Flores
Jurado

MSc. Jorge Luis Sandoval Rivera
Jurado

MSc. Juan Manuel Jacinto Sandoval
Jurado

Dedicatoria:

Agradezco a Dios y a la Buena Madre por permitirme alcanzar este gran sueño.

Dedico este proyecto a mis padres Pedro y Soledad por su gran amor y motivación, A mi amada Flor y a nuestro esperado Bebé, A mis hermanas y sobrinos, y a todas las personas que han sido parte fundamental para llevar a cabo este proyecto,

Br. Pedro David Saavedra Puchulán

Ejecutor

Dedico este proyecto a mis padres Francisco y Edita, por ser los principales promotores de mis sueños, a mi abuela Irma en paz descansa, estaría orgullosa de mí.

Br. José Antonio Burneo Chávez

Ejecutor

Este proyecto de investigación se lo dedico en primer lugar a Dios Padre por darnos la vida y el deseo de seguir adelante, en segundo lugar, a mi padre César que en paz descansa, sé que sentiría orgulloso de mí, a pesar de que ya no está conmigo siempre me cuidará desde el cielo.

Br. Kenmy Montes Torres

Ejecutor

Agradecimiento:

Expresamos nuestro agradecimiento a la Universidad Nacional de Piura por seguir apostando por la educación en nuestro país; a nuestro asesor y maestros por inculcarnos valores e incentivarnos a la investigación.

A nuestras familias por su incondicional apoyo.

ÍNDICE GENERAL

INTRODUCCIÓN.....	1
1. PLANTEAMIENTO DEL PROBLEMA	2
1.1. Situación problemática	2
1.2. DEFINICIÓN DEL PROBLEMA.....	2
2. HIPÓTESIS DE TRABAJO.....	3
3. OBJETIVOS.....	3
3.1. Objetivo general	3
3.2. Objetivos específicos	3
4. JUSTIFICACIÓN	3
5. ALCANCES.....	4
6. DELIMITACIONES.....	4
7. VIABILIDAD	4
7.1. Viabilidad Técnica	4
7.2. Viabilidad Económica	4
7.3. Viabilidad Social.....	5
7.4. Viabilidad Operativa	5
CAPÍTULO I.....	6
MARCO TEÓRICO	6
1.2. BASES TEÓRICAS.....	8
1.2.1 ANSI/TIA942-A.....	8
1.2.3 Shield Ethernet W5100	17
1.2.5 Sensor de presencia PIR	21
1.2.6 Sensor de Humo MQ2.....	23
1.2.7 Buzzer pasivo.....	25
CAPÍTULO II.....	31

METODOLOGÍA.....	31
2.1 Diseño de la Solución.....	31
2.1.1 Diseño de la Infraestructura del proyecto	31
2.1.2 Diagrama esquemático de la Solución.	32
2.2 Planeación del Proyecto.....	33
2.4.1 Hardware y materiales	36
2.4.2 Software.....	36
2.5 Procedimiento Y Casos De Uso De Los Procesos	37
2.6 Procedimiento de armado de maqueta.....	44
2.6.1 Preparación de sensores y del circuito Arduino.....	44
2.6.2 Montaje de los sensores en la maqueta	46
CAPÍTULO III.....	49
PRUEBAS Y RESULTADOS	49
3.1 Prueba de verificación de los parámetros leídos por los sensores	50
3.2 Prueba de visualización grafica de los parámetros obtenidos por los sensores.....	51
3.3 Prueba de recepción de correo electrónico.	52
ANEXOS.....	54
CONCLUSIONES	68
RECOMENDACIONES.....	69
REFERENCIAS BIBLIOGRÁFICAS.....	70
GLOSARIO.....	71

ÍNDICE DE TABLAS

Tabla 01. Directrices Térmicas según ASHRAE	10
Tabla 02. Especificaciones técnicas de Arduino Uno R3	12
Tabla 03. Diagrama de Gantt de las actividades del Proyecto.	33
Tabla 04. Costos de desarrollo del prototipo	34
Tabla 05. Costos de los equipos, materiales y servicios para el prototipo.....	34
Tabla 06. Costo total del prototipo	35
Tabla 07. Requerimiento inicio de sesión.....	37
Tabla 08. Requerimiento finalizar de sesión	37
Tabla 09. Requerimiento cambio de contraseña	38
Tabla 10. Requerimiento crear usuario	38
Tabla 11. Requerimiento eliminar usuario.....	39
Tabla 12. Requerimiento crear zona	39
Tabla 13. Requerimiento eliminar zona.....	40
Tabla 14. Requerimiento editar zona	40
Tabla 15. Requerimiento para Visualizar reporte de eventos	41
Tabla 16. Requerimiento para exportar reporte en formato .xls	41
Tabla 17. Requerimientos no funcionales	42

ÍNDICE DE FIGURAS

Figura 1: Arduino Uno.....	11
Figura 2: Componentes de un Arduino Uno.....	13
Figura 3: Shield Ethernet W5100.....	18
Figura 4: Shield ethernet W5100 acoplado al Arduino Uno.	19
Figura 5: Sensor DHT11.	20
Figura 6: Sensor HC-SR501.....	22
Figura 7: Sensor MQ2.	24
Figura 8: Buzzer Pasivo.	25
Figura 9: Funcionamiento del Buzzer.	26
Figura 10: Componentes de un Buzzer.	26
Figura 11: Componentes de un altavoz.....	27
Figura 12: Disposición de pines de buzzer.	28
Figura 13: Conexión del buzzer al Arduino.	28
Figura 14: Circuito amplificador para altavoz.....	29
Figura 15: Modelo - Vista – Controlador.....	30
Figura 16: Diagrama de Bloques General de la solución propuesta.....	31
Figura 17: Diagrama esquemático de la solución propuesta.	32
Figura 18: Diagrama de caso de uso de la autenticación de usuario	43
Figura 19: Diagrama de caso de uso de inicio y cierre de sesión.	43
Figura 20: Preparación del cableado de los sensores.	44
Figura 21: Circuito impreso donde se conectarán los sensores.....	44
Figura 22: Circuito impreso acoplado como un shield de arduino.....	45
Figura 23: Maqueta que contendrá el circuito.	46
Figura 24: Montado de los sensores en la maqueta.....	47

Figura 25: Maqueta correspondiente al primer circuito Arduino (Sede principal).	47
Figura 26: Maqueta correspondiente al segundo circuito Arduino (Sede secundaria).	48
Figura 27: Diseño del Dashboard del sistema de monitoreo de Data Center.	49
Figura 28: Monitoreo de las lecturas de los sensores mediante el monitor serial.	50
Figura 29: Cuadro de lecturas de los parámetros.	51
Figura 30: Cuadro de reporte de los parámetros.	51
Figura 31: Captura de pantalla de las alertas activas.	52
Figura 32: Captura de pantalla de email de alerta de humo.	53
Figura 33: Diagrama del circuito impreso.	59
Figura 34: Diagrama del circuito impreso con componentes.	59
Figura 35: Diagrama del circuito impreso sin componentes.	60
Figura 36: Diagrama esquemático del circuito impreso.	60
Figura 37: Modelo de la base de datos.	61
Figura 38: Captura de pantalla del login del aplicativo web del sistema de monitoreo.	62
Figura 39: Captura de pantalla del listado de agencias.	63
Figura 40: Captura de pantalla del listado de usuarios.	64
Figura 41: Captura de pantalla del registro de agencias.	65
Figura 42: Captura de pantalla del registro de usuarios.	66
Figura 43: Captura de pantalla del diagrama de Gantt realizado con el programa Gantt Project	67

ÍNDICE DE ANEXOS

ANEXO 1: Código para controlar el Arduino uno.	54
ANEXO 2: Diseño del circuito impreso para la placa del Arduino.....	59
ANEXO 3: Modelado de la Base de Datos.....	61
ANEXO 4: Captura de pantalla del login del aplicativo web.	62
ANEXO 5: Captura de pantalla del listado de agencias.	62
ANEXO 6: Captura de pantalla del listado de usuarios.	64
ANEXO 7: Captura de pantalla del registro de agencias.....	65
ANEXO 8: Captura de pantalla del registro de Usuarios.	66
ANEXO 9: Diagrama de actividades	67

RESUMEN

El propósito de la investigación titulada "Prototipo de sistema de monitoreo de alarmas en una red de data centers" es el mantener la integridad física de los equipos ubicados en este ambiente. Para lo cual es necesario diseñar e implementar un monitoreo de alarmas constante a través de sensores los cuales permitirán monitorear el estado de los Data Center y así poder visualizar los parámetros ambientales como son la temperatura y la humedad relativa del ambiente, también se podrán visualizar parámetros como la presencia de humo y de presencia individuos que ingresan a este ambiente. Esta información se podrá visualizar en tiempo real a través de una página web y también podrán recibir mensajes de texto a los teléfonos celulares que estén configurados para así poder monitorear el estado de las alarmas vigentes.

Este prototipo ayudará en la toma de decisiones para mitigar las posibles fallencias físicas a las cuales son susceptibles los equipos de los Data center, como son el caso de sobrecalentamiento de los equipos por falla del sistema de refrigeración que podría conllevar a que estos se incendien y ocurran daños mayores, también evitar la corrosión por humedad.

Otro beneficio del prototipo es que podrá avisarnos en tiempo real cuando un individuo ingresa al Data center, con esto se podría tener un mejor control de acceso y saber el momento exacto en el cual ingresó a este.

Palabras clave: Prototipo, Sistema de monitoreo, alarmas, sensores, data center.

ABSTRACT

The purpose of the research entitled "Prototype Alarm monitoring system in a data center network" is to maintain the physical integrity of the equipment located in this environment. For this, it is necessary to design and implement a constant alarms monitoring through sensors which will allow to monitor the state of the Data Center and thus be able to visualize the environmental parameters such as the temperature and the relative humidity of the environment, it will also be possible to visualize parameters such as the presence of smoke and presence of individuals entering this environment. This information can be displayed in real time through a Web page and can also receive text messages to cell phones that are configured in order to monitor the status of current alarms.

This prototype will help in the decision making to mitigate the possible physical flaws to which the equipment of the Data center is susceptible, such as the case of overheating of the equipment by failure of the refrigeration system that could lead to that These will fire and occur major damage, also avoid moisture corrosion.

Another benefit of the prototype is that you will be able to notify us in real time when an individual enters the Data center, with this you could have better access control and know the exact moment in which you entered this.

Key words: prototype, monitoring system, alarm, sensors, data center.

INTRODUCCIÓN

En la actualidad la gestión de riesgos se ha vuelto muy importante en las diferentes entidades públicas y privadas, este enfoque estructurado para manejar las probabilidades de ocurrencia de alguna amenaza será tratadas a través de procesos o actividades que nos permitirán prevenir o reducir el impacto negativo del riesgo.

En ocasiones, el manejo de riesgos se orienta en la contención de riesgo por causas físicas o legales. En nuestro caso con la implementación de un prototipo de Sistema de Alarmas de una red de Data centers nos basaremos más en la prevención y toma de decisiones para mitigar estos posibles riesgos físicos (por ejemplo, incendios y acceso no autorizado a las instalaciones, lo que nos permitirá mantener la integridad física de los equipos de una red de data center sabiendo que además de ser costosos son también muy importantes ya que manejar información valiosa y sistemas que la empresa utiliza para sus actividades diarias.

Por tal motivo, mediante la implementación de este prototipo se obtendrán resultados positivos para una mejor toma de decisiones al momento de prevenir potenciales riesgo de tales tipos.

En el capítulo I se presenta el marco teórico de los principales conceptos y especificaciones técnicas, tales como placa Arduino Uno R3, Data Center, sensores, terminales digitales, y otros relacionados a ellos, además se fundamenta los estándares y normas utilizadas, y antecedentes de la investigación.

En el capítulo II se desarrolla la metodología de la investigación, donde se contempla el diseño y los diagramas de la solución, así como materiales, software y hardware usados para la implementación del trabajo de investigación.

En el capítulo III se presenta las pruebas y los resultados obtenidos en el diseño y ejecución del prototipo de sistema de monitoreo de alarmas.

1. PLANTEAMIENTO DEL PROBLEMA

1.1. Situación problemática

La siguiente investigación pretende dar solución a la necesidad de un sistema de monitoreo de alarmas para una red de data centers de la Caja Municipal de Sullana, actualmente solo cuenta con detector de humo, que al detectar la presencia de humo en el data center activa el sistema contra incendios, mas no envía un reporte de incidencia de lo que está sucediendo al personal encargado, por esta razón se va a desarrollar un prototipo de sistema que informe el estado situacional de los parámetros ambientales como la presencia de humo, la temperatura y humedad relativa anormales o fuera de rango así como la presencia de individuos en alguno de los data centers; a través de un dashboard, envío de correos electrónicos y mediante mensaje de texto a los teléfonos celulares del personal autorizado.

1.2. DEFINICIÓN DEL PROBLEMA

En la actualidad las empresas manejan grandes cantidades de información, esta debe ser protegida ya que es uno de los recursos más preciados de la organización, por tal motivo para salvaguardar estos recursos supone salvaguardar los equipos que almacenan estos, además de los equipos de comunicaciones, por tal motivo es que se propone desarrollar un prototipo que monitoree las alarmas de los parámetros de temperatura, humedad relativa, humo y presencia de individuos.

Este sistema deberá ser capaz de cubrir los requerimientos del cliente y a la vez poder ser escalable, es decir, poder agregarle más funcionalidades en un futuro.

2. HIPÓTESIS DE TRABAJO

Es posible y viable diseñar un prototipo de sistema de monitoreo de alarmas en una red de data centers.

3. OBJETIVOS

3.1. Objetivo general

Diseñar un prototipo de sistema de monitoreo de alarmas para una red de data centers.

3.2. Objetivos específicos

- a) Diseñar e implementar el sistema electrónico de lectura de los parámetros a monitorear.
- b) Diseñar e implementar el software del prototipo de monitoreo de alarmas y envío de alertas.
- c) Comprobar la funcionalidad del prototipo del sistema en un ambiente controlado.

4. JUSTIFICACIÓN

A través de un sistema de monitoreo de alarmas se podrá visualizar en tiempo real los estados de las alarmas y los parámetros ambientales de los data centers de la red que tiene la Caja Municipal de Sullana, esto conllevará a una mejora en las políticas de mantenimiento preventivo de los equipos que se encuentran dentro de este ambiente, cabe resaltar que al ser equipos costosos y que almacenan información y servicios, estos deberán funcionar en condiciones ideales para su óptimo desempeño.

5. ALCANCES

La investigación se encuentra enmarcada en el campo del internet de las cosas (IoT) y se realizó para cubrir la necesidad que tenían los data centers de la Caja Municipal de Sullana (Sede Principal ubicada en Sullana) para monitorear el estado de estos y se realizó durante el periodo comprendido entre los meses marzo y julio del 2019.

6. DELIMITACIONES

Este trabajo de investigación plantea la elaboración de un prototipo del sistema de monitoreo de alarmas para una red de data centers, donde se podrá medir parámetros como la presencia de humo, la temperatura, la humedad relativa y presencia de individuos en el data center. Para efectos de nuestra investigación hemos utilizado sensores no industriales, para escenarios reales se recomienda el uso de sensores de mayor precisión.

7. VIABILIDAD

7.1. Viabilidad Técnica

Es viable técnicamente ya que actualmente los data Center de la Caja Municipal de Sullana no cuentan con un sistema de monitoreo de alarmas. Además de que los recursos para su elaboración son fáciles de encontrar y de bajo costo.

7.2. Viabilidad Económica

Al tratarse de un prototipo la inversión que se tomó para su elaboración fue mínima. Los dispositivos que se utilizaron tiene un bajo costo en el mercado.

7.3. Viabilidad Social

La implementación de un sistema de monitoreo de alarmas es viable socialmente ya que generará puestos de trabajo a quienes realicen mantenimiento o mejoras al sistema haciéndolo más escalable.

7.4. Viabilidad Operativa

Es viable operativamente ya que su interfaz web es amigable y cuan práctica y necesaria la recepción de mensajes de texto con las alertas para que el usuario tome decisiones según las lecturas del monitoreo, además de que actualmente la Institución no cuenta con este sistema.

CAPÍTULO I

MARCO TEÓRICO

1.1 ANTECEDENTES DE LA INVESTIGACIÓN

1.1.1. Antecedentes nacionales

En el año 2016 fue presentado en la facultad de Ingeniería y arquitectura, escuela profesional de ingeniería electrónica de la Universidad San Martín de Porres, la tesis “Sistema de Monitoreo de Control Ambiental para Data Center con Alerta vía GSM” por Bonny Lindsay Gonzales Pulido, como requisito para optar el título profesional de ingeniería electrónica.

La investigación se basa en mantener la seguridad de la información con un enfoque ambiental, para el cual diseñaron e implementaron un monitoreo constante por un conjunto de sensores, el cual permite a los dueños o responsables actuar ante algún suceso que perturbe la continuidad operativa del área a monitorear.

La investigación se dedica al monitoreo tanto de la temperatura, humedad y luz, los cuales cuentan cada uno con sensores analógicos y digitales, complementando con módems GSM que permitirá enviar alertas a los responsables, en caso de que los parámetros definidos sobrepasan su valor umbral configurado.

1.1.2. Antecedentes internacionales

En marzo del 2015, se presentó en la facultad de Ingeniería de Sistemas de la Universidad Politécnica Salesiana - Sede Guayaquil la tesis titulada “Análisis, diseño e implementación de un prototipo para un dispositivo de monitoreo ambiental de centros de cómputo de bajo costo basado en tecnología ARM y software de código abierto aplicable a una pequeña y mediana empresa” por Guillermo Sánchez

Herrera y Diana Tintinago Correa, para optar el título profesional de Ingeniero de Sistemas.

El trabajo de desarrollo presenta como objetivo principal diseñar e implementar un sistema web el cual monitorea los factores físicos y ambientales que provocan fallos en los equipos ubicados en un centro de cómputo.

En el año 2018, se presentó en la facultad de ingeniería electrónica de la Universidad Politécnica Salesiana - Sede Guayaquil el proyecto técnico previo a la obtención del título de ingeniero electrónico titulado “Diseño e implementación de un sistema de control de acceso y monitoreo de sensores para Data centers de la empresa Quifatex. s.a, utilizando hardware libre” presentada por los señores Andrey Bishara Suarez López y José Eduardo Muguerza Jaramillo.

El trabajo de desarrollo presenta como objetivo principal diseñar e implementar un sistema web el cual monitorea el control de acceso del personal y los parámetros ambientales de un data center haciendo uso de herramientas de software y hardware libre.

1.2. BASES TEÓRICAS

1.2.1 ANSI/TIA942-A

Es el estándar de calidad creado en abril del 2005, es una norma americana en donde se especifica los requisitos para el diseño de la infraestructura de un data center, la topología que presenta esta norma es aplicable a data center de cualquier tamaño y abarca toda la infraestructura física, así como la ubicación del sitio, el diseño eléctrico, el control de acceso a la red y seguridad, la gestión de base de datos además de la protección contra riesgos físicos, control ambiental, entre otros.

Las especificaciones que se menciona por la asociación de ASHRAE son provistas para pequeños data centers o para data centers que ocupan múltiples pisos o cuartos. TIA-942-A se ha actualizado para modificar los rangos de operación más bajos para la humedad basado en el punto de rocío en lugar de humedad relativa. (s/p)

Además, el estándar TIA-942-A establece que las salas o cuartos de ordenadores deberán cumplir con la clase 1 de la TIA-569-C, que van acorde con las especificaciones técnicas de ASHRAE y son las siguientes:

- Temperatura: 18°C (64,4°F) – 27°C (80,6°F) de bulbo seco.
- Humedad relativa máxima: 60%.
- Humedad mínima: 5,5°C (41,9°F).
- Punto máximo de rocío: 15 ° C (59 °F).
- La velocidad máxima del cambio de temperatura: 5 °C (9 °F) por hora.

Los rangos permitidos para la clase 1 son de 15°C a 32°C con una humedad relativa de 20% a 80% mientras que para la clase 2 los rangos permitidos son 10°C a 35°C y de 20% a 80% de humedad relativa. Se recomienda que los operadores de los data center deben mantener estos parámetros para así poder evitar problemas en los equipos, se sabe que, a mayor temperatura, menor humedad relativa y esto puede conllevar a que los equipos sufran descargas electrostáticas. Por otro lado, si los data center tienen una humedad alta esto generaría corrosión en los equipos y por ende se pueden dañar y sufrir pérdida de información.

A continuación, se mostrará una tabla que contiene las directrices térmicas según ASHRAE.

clases	Especificación de Equipo ambiental							
	Operación de productos (a)(b)				Apagado de productos (b)(c)'			
	Temperatura de Bulbo seco (°C) (d)(f)	Ámbito de humedad, sin condensación(g)(h)	Punto de rocío máximo (°C)	Elevación máxima (m)	Velocidad máxima de cambio (°C/hr)(e)	Temperatura de bulbo seco (°C)	Humedad relativa (%)	Máximo punto de rocío (°C)
	Recomendado (se aplica a todas las clases A, los centros de datos individuales pueden optar por ampliar este rango basado en el análisis descrito en este documento)							
A1 a A4	18 a 27	-9 °C DP a 15 °C DP y a 60% RH						
Admisible								
A1	15 a 32	-12 °C DP & 8% RH a 17° C DP & 80% RH	17	3050	5/20'	5 a 45	8 a 80	27
A2	10 a 35	-12 °C DP & 8% RH a 21° C DP & 80% RH	21	3050	5/20'	5 a 45	8 a 80	27
A3	5 a 40	-12 °C DP & 8% RH a 24° C DP & 85% RH	24	3050	5/20'	5 a 45	8 a 85	27
A4	5 a 45	-12 °C DP & 8% RH a 24° C DP & 90% RH	24	3050	5/20'	5 a 45	8 a 90	27
B	5 a 35	8% a 28° C DP & 80% RH	28	3050	NA	5 a 45	8 a 90	29
C	5 a 40	8% a 28° C DP & 80% RH	28	3050	NA	5 a 45	8 a 90	29

Tabla 01. Directrices Térmicas según ASHRAE.

Fuente: ASHRAE

1.2.2 Placa Arduino Uno R3

“Arduino Uno es una placa electrónica basada en el microcontrolador ATmega328. Cuenta con 14 entradas/salidas digitales, de las cuales 6 se pueden utilizar como salidas PWM (Modulación por ancho de pulsos) y otras 6 son entradas analógicas. Además, incluye un resonador cerámico de 16 MHz, un conector USB, un conector de alimentación, una cabecera ICSP y un botón de reseteo. La placa puede ser alimentada mediante un cable USB si se conecta a un ordenador o a través de una fuente de poder de 9V”. (Jadiaz, 2016)



Figura 1: Arduino Uno.

Fuente: <http://www.iescamp.es/miarduino/>

Especificaciones técnicas de Arduino Uno r3

Especificaciones Técnicas	
Microcontrolador	ATmega328
Voltaje de funcionamiento	5V
Voltaje de entrada (recomendado)	7-12V
Voltaje de entrada (límites)	6-20V
Pines Digitales I/O	14(de los cuales 6 proporcionan salida PWM)
Pines Digitales I/O PWM	6
Pines de entrada analógicos	6
DC Current por pines de I/O	20 mA
DC Current por pines de 3.3V	50 mA
Memoria Flash	32 KB (ATmega328) de los cuales 0.5 KB son utilizados para el arranque
SRAM	2 KB (ATmega328)
EEPROM	1 KB (ATmega328)
Velocidad de Reloj	16 MHz
Longitud	68.6 mm
Ancho	53.4 mm
Peso	25 g

Tabla 02. Especificaciones técnicas de Arduino Uno R3.

Fuente: <http://www.iescamp.es/miarduino/>

“Podemos identificar los componentes de un Arduino fácilmente de la siguiente manera, si miramos la placa desde la parte de arriba, así como se muestra en la siguiente figura del esquema se podrán observar los componentes empezando según las agujas del reloj, es decir, en sentido horario”. (Jadiaz, 2016)

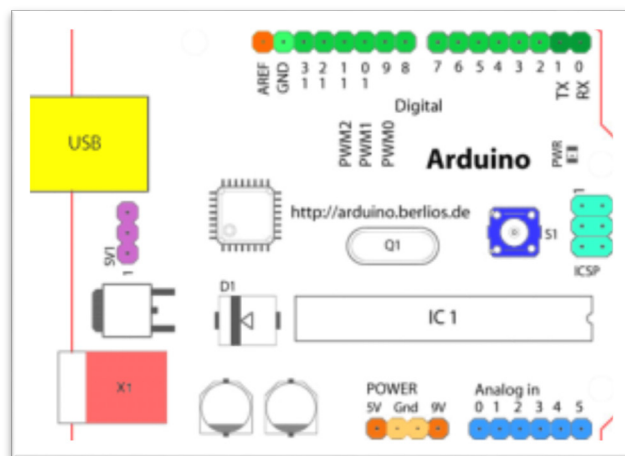


Figura 2: Componentes de un Arduino Uno.

Fuente: <http://www.iescamp.es/miarduino/>

- Terminal de referencia analógica (naranja)
- Tierra digital (verde claro)
- Terminales digitales 2-13 (verde)
- Terminales digitales 0-1/ E/S serie – TX/RX (verde oscuro) – Estos pines no se pueden utilizar como e/s digitales (`digitalRead()` y `digitalWrite()`) si estás utilizando comunicación serie (por ejemplo `Serial.begin()`).
- Botón de reinicio – S1 (azul oscuro)
- Programador serie en circuito «In-circuit Serial Programmer» o «ICSP» (azul celeste).

- Terminales de entrada analógica 0-5 (azul claro)
- Terminales de alimentación y tierra (alimentación: naranja, tierras: naranja claro)
- Entrada de alimentación externa (9-12VDC) – X1 (rosa)
- Selector de alimentación externa o por USB (coloca un jumper en los dos pines más cercanos de la alimentación que quieras) – SV1 (púrpura). En las versiones nuevas de Arduino la selección de alimentación es automática por lo que puede que no tengas este selector.
- USB (utilizado para subir programas a la placa y para comunicaciones serie entre la placa y el ordenador; puede utilizarse como alimentación de la placa) (amarillo).

Entradas y salidas digitales/analógicas

“La placa Arduino al ser un dispositivo electrónico posee un conjunto de entradas y salidas, por tanto, este dispositivo acepta información en forma de señal de entrada (input) además desarrolla ciertas operaciones sobre ésta y luego produce señales de salida (output).” (Jadiaz, 2016)

“En los sistemas electrónicos, una magnitud física variable se representa generalmente mediante una señal eléctrica que varía de manera tal que describe esa magnitud. Por lo general, se hace referencia a las señales continuas como señales analógicas, mientras que asociamos las señales discretas a señales digitales: el ejemplo más claro es el de las señales binarias, donde las señales sólo pueden tomar dos niveles, 0 o 1”. (Jadiaz, 2016)

“Arduino incorpora terminales digitales (señales discretas) pero de tal forma que tenemos un gran abanico de valores con los que trabajar (por ejemplo,

255 valores de luz en un fotosensor, siendo 0 ausencia de luz y 254 el máximo valor lumínico)". (Jadiaz, 2016)

Terminales Digitales

"Las terminales digitales de una placa Arduino pueden ser utilizadas para entradas o salidas de propósito general a través de los comandos de programación `pinMode()`, `digitalRead()`, y `digitalWrite()`. Cada terminal tiene una resistencia pull-up que puede activarse o desactivarse utilizando `digitalWrite()` (con un valor de HIGH o LOW, respectivamente) cuando el pin está configurado como entrada. La corriente máxima por salida es 40 mA". (Jadiaz, 2016)

"Serial: 0 (RX) y 1 (TX). Utilizado para recibir (RX) y transmitir (TX) datos serie TTL. En el Arduino Diecimila, estas terminales están conectadas a las correspondientes patas del circuito integrado conversor FTDI USB a TTL serie. En el Arduino BT, están conectados a las terminales correspondientes del módulo Bluetooth WT11. En el Arduino Mini y el Arduino LilyPad, están destinados para el uso de un módulo serie TTL externo (por ejemplo, el adaptador Mini-USB)". (Jadiaz, 2016)

"**Interruptores externos:** 2 y 3. "Estas terminales pueden ser configuradas para disparar una interrupción con un valor bajo, un pulso de subida o bajada, o un cambio de valor. Mira la función `attachInterrupt()` para más detalles". (Jadiaz, 2016)

"**PWM: 3, 5, 6, 9, 10, y 11.** Proporcionan salidas PWM de 8 bit con la función `analogWrite()`. En placas con ATmega8, las salidas PWM solo están disponibles en los pines 9, 10, y 11". (Jadiaz, 2016)

"**Reset BT:** 7. (solo en Arduino BT) Conectado a la línea de reset del módulo bluetooth". (Jadiaz, 2016)

“**SPI**: 10 (SS), 11 (MOSI), 12 (MISO), 13 (SCK). Estas terminales soportan comunicación SPI. Aunque esta funcionalidad está proporcionada por el hardware, no está incluida actualmente en el lenguaje Arduino”. (Jadiaz, 2016)

“**LED**: 13. En el Diecimila y el LilyPad hay un led en placa conectado al pin digital 13. cuando el pin tiene valor HIGH, el LED está encendido, cuando el pin está en LOW, está apagado”. (Jadiaz, 2016)

Pines Analógicos

“Los pines de entrada analógicos soportan conversiones analógico-digital (ADC) de 10 bit utilizando la función **analogRead** (). Las entradas analógicas pueden ser también usadas como pines digitales: entrada analógica 0 como pin digital 14 hasta la entrada analógica 5 como pin digital 19. Las entradas analógicas 6 y 7 (presentes en el Mini y el BT) no pueden ser utilizadas como pines digitales”. (Jadiaz, 2016)

“**I2C**: 4 (SDA) y 5 (SCL). Soportan comunicaciones I2C (TWI) utilizando la librería Wire (documentación en la página web de Wiring)”. (Jadiaz, 2016)

Pines de alimentación

“**VIN** (a veces marcada como «9V»). Es el voltaje de entrada a la placa Arduino cuando se está utilizando una fuente de alimentación externa (En comparación con los 5 voltios de la conexión USB o de otra fuente de alimentación regulada). Puedes proporcionar voltaje a través de este pin. Date cuenta que diferentes placas aceptan distintos rangos de voltaje de entrada, por favor, mira la documentación de tu placa. También date cuenta que el LilyPad no tiene pin VIN y acepta solo una entrada regulada”. (Jadiaz, 2016)

“**5V**. La alimentación regulada utilizada para alimentar el microcontrolador y otros componentes de la placa. Esta puede venir de VIN a

través de un regulador en placa o ser proporcionada por USB u otra fuente regulada de 5V". (Jadiaz, 2016)

3V3. Una fuente de 3.3 voltios generada por el chip FTDI de la placa.

GND. Pines de tierra.

Otros Pines

"AREF. Referencia de voltaje para las entradas analógicas. Utilizada con la función `analogReference()`.

Reset. Pon esta línea a LOW para resetear el microcontrolador. Utilizada típicamente para añadir un botón de reset a shields que bloquean el de la placa principal". (Jadiaz, 2016)

1.2.3 Shield Ethernet W5100

"El shield posee un conector ethernet estándar RJ45. Este shield dispone de unos conectores que permiten conectar a su vez otras placas encima y apilarlas sobre la placa Arduino". (Jadiaz, 2016)

"Arduino usa los pines digitales 10, 11, 12, y 13 (SPI) para comunicarse con el W5100 en el ethernet shield. Estos pines no pueden ser usados para entradas o salidas genéricas". (Jadiaz, 2016)

"El botón de reset del shield resetea ambos, el W5100 y la placa Arduino." (Jadiaz, 2016)

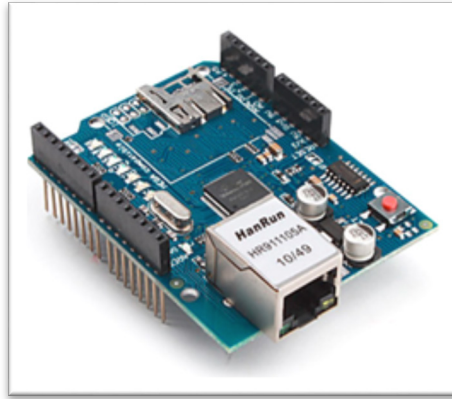


Figura 3: Shield Ethernet W5100.

Fuente: <http://www.iescamp.es/miarduino/>

El shield contiene LEDs para información:

- ON: indica que la placa y el shield están alimentadas.
- LINK: indica la presencia de un enlace de red y parpadea cuando el shield envía o recibe datos.
- 100M: indica la presencia de una conexión de red de 100 Mb/s (de forma opuesta a una de 10Mb/s).
- RX: parpadea cuando el shield recibe datos.
- TX: parpadea cuando el shield envía datos.

“El jumper soldado marcado como “INT” puede ser conectado para permitir a la placa Arduino recibir notificaciones de eventos por interrupción desde el W5100, pero esto no está soportado por la librería Ethernet. El jumper conecta el pin INT del W5100 al pin digital 2 de Arduino”. (Jadiaz, 2016)

Para encajar el shield en nuestro Arduino:

- Verificar que el Arduino apagado.
- Los pines tienen que encajar con suavidad primero, y después presionar cuidadosamente para que se una a Arduino. Sin forzarlo.
- Asegurarnos de que todos y cada uno de los pines encaja en los conectores de abajo.

“Una vez que haya encajado se puede alimentar el Arduino y luego conectar el cable de red, debería tener un aspecto similar a este, y se observará el encendido de varios de LEDs”. (Jadiaz, 2016)



Figura 4: Shield ethernet W5100 acoplado al Arduino Uno.

Fuente: <http://www.iescamp.es/miarduino/>

Con el cable Ethernet con RJ45 en ambos extremos, uno de ellos va al Shield y el otro al Router o Switch de que dispongas.

1.2.4 Sensor de temperatura y humedad (Sensor DHT11)

Según Naylamp Mechatronics “El DHT11 es un sensor digital de temperatura y humedad relativa de bajo costo y fácil uso. Integra un sensor capacitivo de humedad y un termistor para medir el aire circundante, y muestra los datos mediante una señal digital en el pin de datos (no posee salida analógica). Utilizado en aplicaciones académicas relacionadas al control automático de temperatura, aire acondicionado, monitoreo ambiental en agricultura y más”. (NM, s.f)

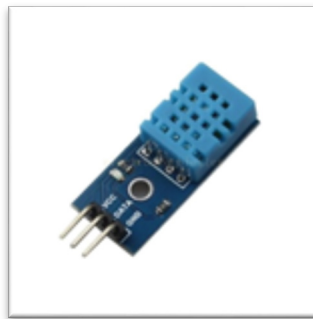


Figura 5: Sensor DHT11.

Fuente: <https://naylampmechatronics.com>

“Utilizar el sensor DHT11 con las plataformas Arduino/Raspberry Pi/Nodemcu es muy sencillo tanto a nivel de software como hardware. A nivel de software se dispone de librerías para Arduino con soporte para el protocolo "Single bus". En cuanto al hardware, solo es necesario conectar el pin VCC de alimentación a 3-5V, el pin GND a Tierra (0V) y el pin de datos a un pin digital en nuestro Arduino. Si se desea conectar varios sensores DHT11 a un mismo Arduino, cada sensor debe tener su propio pin de datos. Quizá la única desventaja del sensor es que sólo se puede obtener nuevos datos cada 2 segundos. Cada sensor es calibrado en fabrica para obtener unos coeficientes de calibración grabados en su memoria OTP, asegurando alta estabilidad y

fiabilidad a lo largo del tiempo. El protocolo de comunicación entre el sensor y el microcontrolador emplea un único hilo o cable, la distancia máxima recomendable de longitud de cable es de 20m., de preferencia utilizar cable apantallado”. [NM] (s.f)

- Voltaje de Operación: 3V - 5V DC
- Rango de medición de temperatura: 0 a 50 °C
- Precisión de medición de temperatura: ± 2.0 °C
- Resolución Temperatura: 0.1°C
- Rango de medición de humedad: 20% a 90% RH.
- Precisión de medición de humedad: 5% RH.
- Resolución Humedad: 1% RH
- Tiempo de censado: 1 seg.
- Interface digital: Single-bus (bidireccional)
- Modelo: DHT11
- Dimensiones: 16*12*5 mm
- Peso: 1 gr.
- Carcasa de plástico celeste

1.2.5 Sensor de presencia PIR

“El Sensor PIR **HC-SR501** ayuda a captar movimiento, casi siempre se utiliza para detectar si un ser humano se ha movido dentro o fuera de la gama de sensores. Es pequeño, de bajo costo, bajo consumo de energía además de ser fácil de usar y no sufrir desgaste. Por tal motivo este tipo de sensor se puede encontrar comúnmente en los electrodomésticos y aparatos utilizados en los hogares o negocios”. Nextiafenix. (s.f).

“Los sensores PIR también son comúnmente llamados **sensores de movimiento IR o piroeléctricos**. Los sensores PIR son básicamente un sensor piroeléctrico, que puede detectar los niveles de radiación infrarroja. Cualquier cosa emite cierta radiación de bajo nivel, entre más caliente esté, más radiación emite. El sensor de movimiento PIR **Hc-SR501** es capaz de detectar un cambio en los niveles de IR promedio de su zona de detección (por ejemplo, cuando un humano entra en una habitación, esa habitación se incrementará en temperatura ligeramente) y por lo tanto detectará el movimiento”. Nextiafenix. (s.f).

“**El HC-SR501** es compatible directamente con la tecnología TTL y por lo tanto también con la mayoría de microcontroladores comerciales como Arduino, PIC, AVR, COP, DSP, STM32, 8051, etc. puede ser alimentado desde 5V a 20V y tiene una salida digital de 3.3V”. Nextiafenix. (s.f).

“El HC-SR501 incluye un retardo ajustable antes de disparar (aproximadamente 0,5 a 200 segundos) y tiene una sensibilidad ajustable y dos orificios de montaje M2. Su rango de detección es de hasta **7 metros** con un ángulo de **120 grados**. Es un gran sensor para proyectos de robótica y aplicaciones de automatización del hogar en el que se requiere detectar cambios en el medio ambiente y en especial movimientos de un ser humano. Este módulo tiene un consumo de energía muy bajo sobre **50μA**, por lo que es especialmente adecuado para productos de alimentación por batería”. Nextiafenix. (s.f).



Figura 6: Sensor HC-SR501.

Fuente: <https://www.nextiafenix.com/>

Características de HC-SR501 sensor PIR detección de movimiento

- Dimensiones aproximadas: 3.2cm x 2.4cm x 1.8 cm
- Circuitos de control incluidos en el módulo
- Tiempo de activación y sensibilidad ajustable
- Voltaje de operación: 4.5 – 20 V
- Corriente de operación: <60uA
- Salida: Pulso lógico 3.3V
- Tiempo de retardo: 5 – 200 Segundos (ajustable)
- Distancia de detección: 3 – 7 Metros (ajustable)

1.2.6 Sensor de Humo MQ2

“El módulo sensor de gas MQ-2 es un dispositivo que nos puede ayudar a detectar concentraciones de diferentes tipos de gas como lo son Gas Licuado de Petróleo (GLP), Butano, Metano, Alcohol, Hidrógeno, Humo, Isobutano, Monóxido de Carbono y Propano”. Nextiafenix. (s.f).

“El sensor de gas MQ-2 presenta una respuesta rápida y estable también contiene en su interior una resistencia para su protección además de un preset para ajustar la sensibilidad y el cual afecta el disparo de la salida **DO** del módulo”. Nextiafenix. (s.f).

“Puede ser utilizado en un sin fin de aplicaciones solo para dar algunos ejemplos tenemos, alarma contra incendios gracias a que es sensible también al humo, puede ser utilizado en la prevención de fugas de gas, o para mantener determinada concentración de gas en un área, característica muy útil por ejemplo en minas donde se presenta cierta cantidad de gases los cuales no deben pasar un nivel determinado, sensor para corroborar la combustión eficiente de gas en estufas, calentadores, etc”. Nextiafenix. (s.f).

Características del MQ-2 Sensor de Gas:

- Voltaje de Operación adecuado: 5V DC
- Respuesta rápida y alta sensibilidad
- Rango de detección: 300 a 10000 ppm
- Gas característico: 1000 ppm, Isobutano
- Resistencia de censado: 1K Ω 50 ppm Tolueno a 20K Ω in
- Tiempo de Respuesta: $\leq 10s$
- Tiempo de recuperación: $\leq 30s$
- Temperatura de trabajo: -20 °C ~ +55 °C
- Humedad: $\leq 95\%$ RH
- Contenido de oxígeno ambiental: 21%
- Consume menos de 150 mA a 5V.

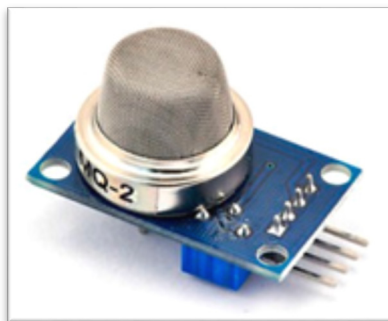


Figura 7: Sensor MQ2.

Fuente: <https://www.nextiafenix.com/>

1.2.7 Buzzer pasivo

“Un buzzer pasivo o un altavoz son dispositivos que permiten convertir una señal eléctrica en una onda de sonido. Estos dispositivos no disponen de electrónica interna, por lo que tenemos que proporcionar una señal eléctrica para conseguir el sonido deseado”. (Llamas, s.f)

“En oposición, los buzzer activos disponen de un oscilador interno, por lo que únicamente tenemos que alimentar el dispositivo para que se produzca el sonido”. (Llamas, s.f).

“Pese a tener la complejidad de proporcionar y controlar nosotros la señal eléctrica, los buzzer pasivos y de los altavoces tienen la ventaja de que podemos variar el tono emitido modificando la señal que aplicamos al altavoz, lo que nos permite generar melodías”. (Llamas, s.f).

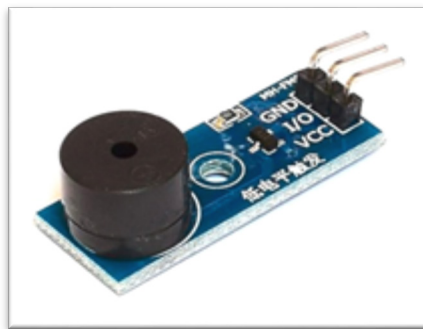


Figura 8: Buzzer Pasivo.

Fuente: <https://www.luisllamas.es/>

Funcionamiento de un Buzzer:

“Técnicamente tanto buzzers como altavoces son transductores electroacústicos, es decir, dispositivos que convierten señales eléctricas en sonido. La diferencia entre ambos es el fenómeno en el que basan su funcionamiento”. (Llamas, s.f).

“Los buzzer son transductores piezoeléctricos. Los materiales piezoeléctricos tienen la propiedad especial de variar su volumen al ser atravesados por corrientes eléctricas”. (Llamas, s.f).

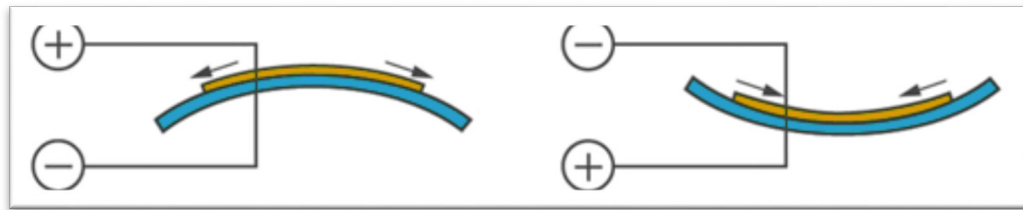


Figura 9: Funcionamiento del Buzzer.

Fuente: <https://www.luisllamas.es/>

“Un buzzer aprovecha este fenómeno para hacer vibrar una membrana al atravesar el material piezoeléctrico con una señal eléctrica”. (Llamas, s.f).

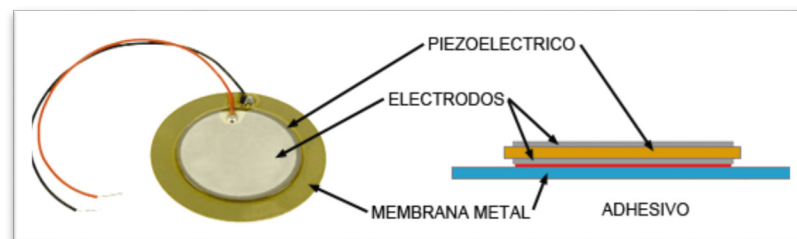


Figura 10: Componentes de un Buzzer.

Fuente: <https://www.luisllamas.es/>

“Los buzzer son dispositivos pequeños y compactos, con alta durabilidad, y bajo consumo eléctrico. Por contra, la calidad de sonido es reducida”. (Llamas, s.f).

“Por su parte, un altavoz basa su funcionamiento en el magnetismo. Se dispone de un imán permanente que, normalmente, es fijo a la carcasa. Por otro lado, una bobina móvil se acopla a una membrana flexible”. (Llamas, s.f).

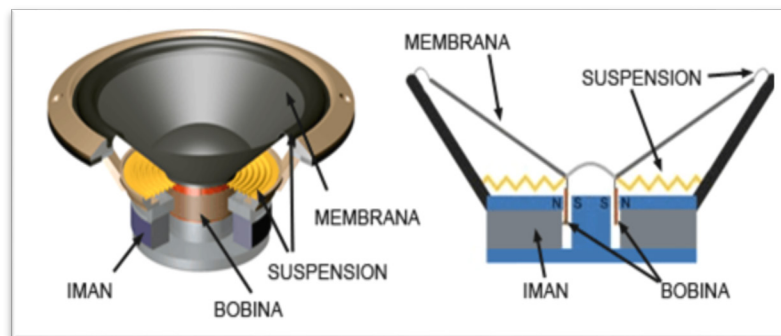


Figura 11: Componentes de un altavoz.

Fuente: <https://www.luisllamas.es/>

“Al hacer circular una corriente por la bobina el campo magnético resultante genera una atracción con el imán, haciendo vibrar la membrana”. (Llamas, s.f).

“Los altavoces, por presentan una mejor calidad de sonido, pero, en general, necesitan mayor potencia y es necesario disponer de dispositivos de amplificación para su uso”. (Llamas, s.f).

Esquema de montaje

“Si usamos una de las placas comerciales para pequeños proyectos y hobbies, que incorporan la electrónica y terminales necesarios, la conexión con Arduino es realmente sencilla. Simplemente alimentamos el módulo conectando

Vcc y GND a Arduino, y la entrada de señal a cualquier salida digital de Arduino".
(Llamas, s.f).

El esquema de conexión visto desde el componente sería el siguiente

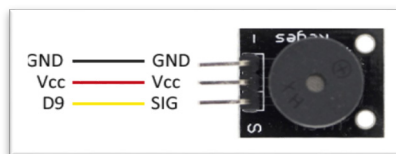


Figura 12: Disposición de pines de buzzer.

Fuente: <https://www.luisllamas.es/>

Mientras que el esquema de conexión visto desde Arduino quedaría así:

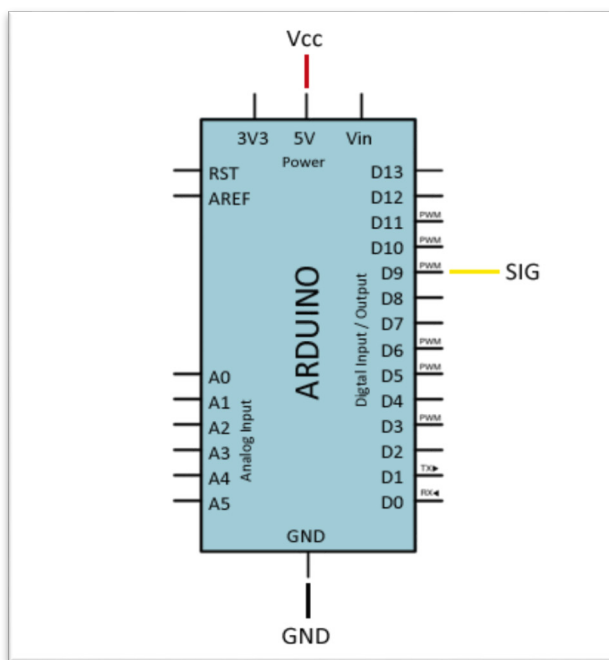


Figura 13: Conexión del buzzer al Arduino.

Fuente: <https://www.luisllamas.es/>

“Si queremos usar un altavoz, que consumen mayor corriente de la que puede proporcionar Arduino, tendremos que proporcionar una etapa de amplificación, como vimos en la entrada de transistores BJT”. (Llamas, s.f).

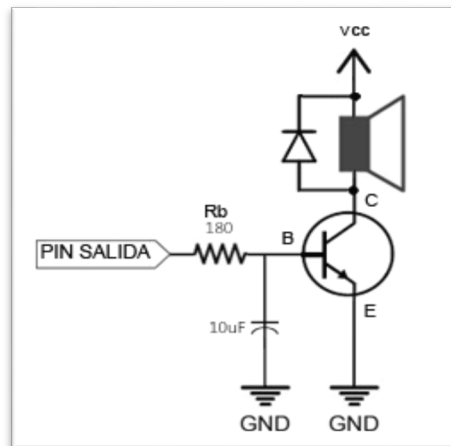


Figura 14: Circuito amplificador para altavoz.

Fuente: <https://www.luisllamas.es/>

“Aunque en general, en estos casos lo normal es que empleemos un amplificador específico diseñado para audio, en lugar de hacerlo nosotros mismos. Conseguiremos mejores niveles de calidad e incluso un menor coste”. (Llamas, s.f).

Especificaciones Técnicas:

- Voltaje de Operación: 3.3V - 5V DC
- Tipo: Piezoeléctrico pasivo
- Incluye el transistor S8550
- Pines: VCC, GND y Señal

1.2.8. MVC

Según el libro “Pro PHP MVC” por Chris Pitt define al MVC (Modelo - Vista - Controlador) como al patrón de diseño de software construido alrededor de la interconexión de tres principales tipos de componentes, en un lenguaje de programación como PHP, a menudo con un fuerte enfoque en Programación orientado a objetos (POO) paradigmas de software. El modelo es donde se guarda toda la lógica empresarial de una aplicación. La aplicación debe acceder a la información en una base de datos, el código para hacerlo se mantendría en el modelo.

La vista es donde se guardan todos los elementos de la interfaz de usuario de nuestra aplicación. Esto puede incluir nuestro marcado HTML, hojas de estilo CSS, y archivos JavaScript. Todo lo que un usuario ve o interactúa con él puede mantenerse en una vista, y a veces, lo que el usuario ve es en realidad una combinación de muchas vistas diferentes en la misma solicitud.

El controlador es el componente que conecta modelos y vistas juntos. Controladores aíslan el negocio.

Los controladores son el primer punto de entrada en este trío de componentes, porque la solicitud se pasa primero a un controlador, que luego creará una instancia de los modelos y las vistas requeridas para cumplir con una solicitud.

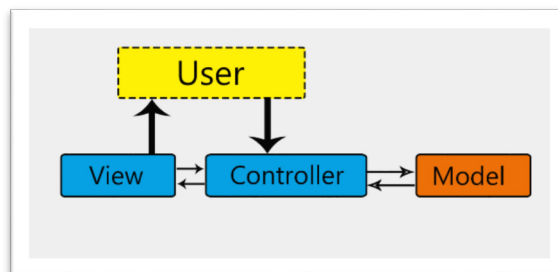


Figura 15: Modelo - Vista – Controlador.

Fuente: Los autores

CAPÍTULO II

METODOLOGÍA

2.1 Diseño de la Solución

2.1.1 Diseño de la Infraestructura del proyecto

El diseño del proyecto se planteó, de acuerdo a la problemática que tenían los data center se vio la necesidad de la implementación de sensores de temperatura, humedad relativa, humo y de presencia de individuos para la toma de decisiones respectivas. Además, para procesamiento de esta información recogida por los sensores se vio la necesidad de utilizar una placa Arduino Uno y para la comunicación entre el Arduino y la red se implementó también un shield ethernet. Por otro lado, se corroboró que la distribución de los puertos del Arduino y shield ethernet estén de acuerdo a los requerimientos de conectividad de cada uno de los sensores de medición y comunicación. Respetando así los parámetros de voltaje, frecuencia y protocolos de transmisión de data.

A continuación, se observa un diagrama de bloques en donde se representa el planteamiento de la solución propuesta.

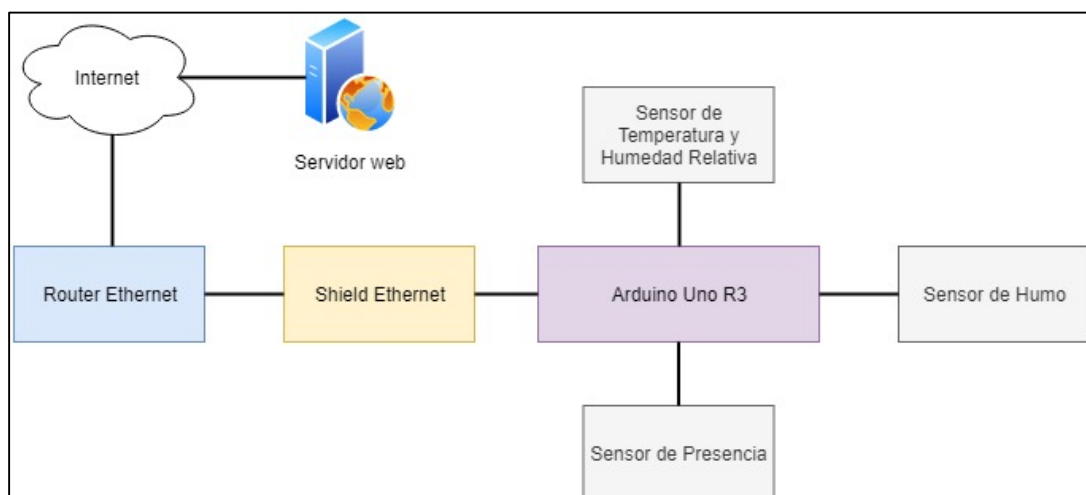


Figura 16: Diagrama de Bloques General de la solución propuesta.

Fuente: Los autores.

2.1.2 Diagrama esquemático de la Solución.

Para la solución a implementar consideraremos como sensores de temperatura y humedad relativa al sensor DTH11 mientras que, para el sensor de humo, el MQ2 que es muy sensible a los gases y CO2. Para la detección de individuos que ingresan al data center hemos utilizado el sensor PIR HC-SR501 y por último para la comunicación entre el Arduino y la red utilizamos el Shield Ethernet W5100, esquematizados en la siguiente figura.

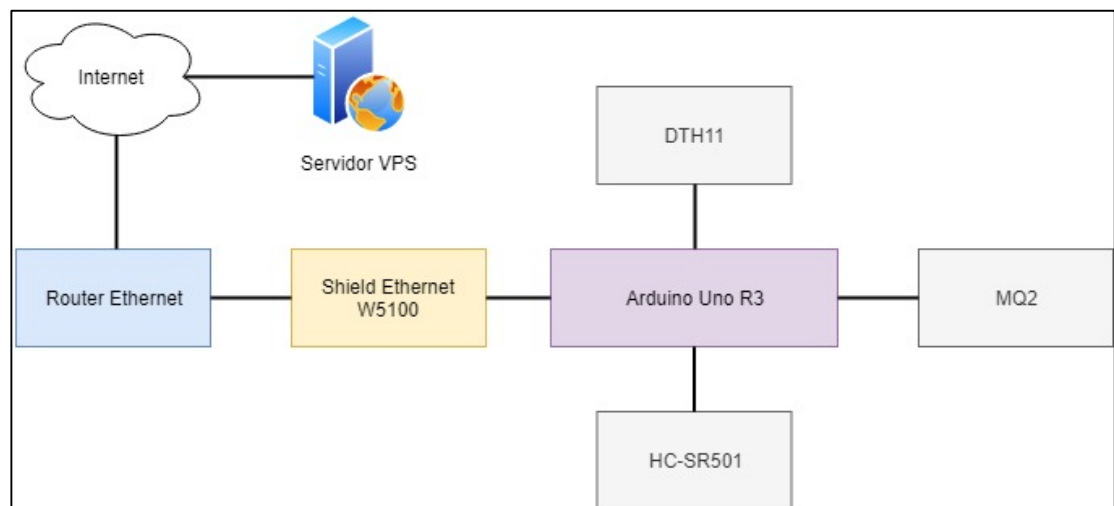


Figura 17: Diagrama esquemático de la solución propuesta.

Fuente: Los autores.

2.2 Planeación del Proyecto

El proyecto tiene una duracion estimada de 4 meses, en las cuales se desarrollarán las siguientes actividades con su respectiva duración.

Actividades	Marzo				Abril				Mayo				Junio				Julio			
	1era	2da	3era	4ta	1era	2da	3era	4ta	1era	2da	3era	4ta	1era	2da	3era	4ta	1era	2da	3era	4ta
Recolección de Información			X	X																
Programación de Arduino					X	X														
Diseño y elaboración del circuito impreso							X	X												
Diseño y elaboración de la Base de datos							X	X	X	X										
Diseño y elaboración de Maqueta								X	X	X										
Diseño e Implementación aplicativo web										X	X	X	X							
Configuración del Servidor VPS														X	X					
Documentación															X	X	X			

Tabla 03. Diagrama de Gantt de las actividades del Proyecto.

Fuente: Los autores

2.3 Presupuesto del Proyecto

2.3.1 Presupuesto de Desarrollo.

Cantidad	Recurso Humano	Costo
1	Analista de Base de Datos	S/2,590.00
1	Programador	S/2,950.00
1	Técnico Electrónico	S/1,880.00
Sub Total		S/7,420.00

Tabla 04. Costos de desarrollo del prototipo.

Fuente: Los autores

En la tabla 04. Se observa los costos de desarrollo del diseño e implementación del prototipo.

2.3.2 Presupuesto de Equipos, materiales y servicios.

Cantidad	Recurso	Costo unit	Costo Total
2	Arduino Uno	S/40.00	S/80.00
2	Shield Ethernet	S/40.00	S/80.00
2	Sensor DTH11	S/10.00	S/20.00
2	Sensor de Movimiento HC-SR501	S/8.00	S/16.00
2	Sensor de Humo MQ2	S/10.00	S/20.00
2	Fuente de Alimentación	S/12.00	S/24.00
2	Cables Patch cord	S/5.00	S/10.00
1	Router Ethernet	S/130.00	S/130.00
1	Placa de fibra para circuito	S/2.00	S/2.00
5 m	Soldadura de estaño	S/5.00	S/5.00
1	Cautín	S/17.00	S/17.00
1	pasta para soldar	S/1.50	S/1.50
120 ml	Óxido Férrico	S/2.00	S/2.00
0.5 m	Triplay	S/10.00	S/10.00
0.5 kg	Cola sintética	S/4.00	S/4.00
50	Clavos 1"	S/2.00	S/2.00
1	Alquiler de servidor VPS	S/71.95	S/71.95
1	Compra de dominio	S/4.30	S/4.30
Subtotal			S/499.75

Tabla 05. Costos de los equipos, materiales y servicios para el prototipo.

Fuente: Los autores

En la tabla 05. Se observa el costo de los equipos que se adquirirán para la elaboración del prototipo, así como de los materiales para la elaboración del circuito y maqueta y finalmente el costo de alquiler de un servidor vps y un dominio.

2.3.3 Presupuesto total

Recursos	Costo
Costo de Recursos Humano	S/7,420.00
Costo de Equipos, materiales y servicios	S/499.75
Total	S/7,919.75

Tabla 06. Costo total del prototipo.

Fuente: Los autores

En la Tabla 06. Se observa el costo total de para la elaboración del prototipo, el monto para realizar el proyecto es de S/. 7,919.75 soles.

2.4 Material y Métodos

2.4.1 Hardware y materiales

- 02 Arduino Uno R3.
- 02 Shield Ethernet W5100.
- 02 Sensor de Temperatura y Humedad relativa DTH11.
- 02 Sensor de Movimiento PIR HC-SR501.
- 02 Sensor de Humo MQ2.
- 02 Fuentes de alimentación de 9V.
- 02 cable patch cord utp directos.
- 01 Router ethernet.
- 01 placa de baquelita para el diseño de la placa de circuito impreso.
- Insumos y herramientas para la elaboración de la maqueta en general.

2.4.2 Software

- Arduino IDE para la codificación y la subida de este al Arduino.
- Lenguaje de Programación: Java y php.
- Lenguaje de marcado: Html.
- Lenguaje de Diseño: css
- Aplicativo Eagle 7.6 para la elaboración de la placa de circuito impreso.
- Utilización de un dominio web y un servidor virtual privado (VPS) para el alojamiento del aplicativo web.

2.5 Procedimiento Y Casos De Uso De Los Procesos

2.5.1 Requerimientos Funcionales

2.5.1.1 Módulo Usuarios

Requerimiento	Inicio de sesión
Descripción	Los usuarios que se encuentran registrados por el administrador, deben ingresar con su usuario y contraseña válidas; si estas son correctas, muestra el dashboard como pantalla de inicio.
Prioridad	Alta
Precondición	<ul style="list-style-type: none">• El usuario y la contraseña deben estar creados en la base de datos.
Entrada	<ul style="list-style-type: none">• Nombre de Usuario• Contraseña
Salida	<ul style="list-style-type: none">• Dashboard• Menú habilitado al usuario
Procedimiento	El usuario ingresa sus credenciales en la pantalla de inicio. El sistema verifica con la base de datos si las credenciales son correctas, en caso lo sea ingresa al sistema mostrando como ventana de inicio el dashboard del sistema, caso contrario el sistema vuelve a mostrar la página de inicio con el mensaje indicando el error.
Postcondición	El usuario ingresa al sistema

Tabla 07. Requerimiento inicio de sesión.

Fuente: Los autores

Requerimiento	Finalizar Sesión
Descripción	El sistema debe permitir que el usuario que esté dentro del sistema, pueda cerrar su sesión
Prioridad	alta
Precondición	El usuario debe estar autenticado en el sistema
Entrada	<ul style="list-style-type: none">• Usuario actual
Salida	Pantalla de inicio de sesión.
Procedimiento	En la navegación del sistema, el usuario debe tener un segmento "usuario" el cual muestre la opción para cerrar sesión existente.
Postcondición	La sesión actual finaliza.

Tabla 08. Requerimiento finalizar de sesión.

Fuente: Los autores

Requerimiento	Cambiar Contraseña
Descripción	El usuario puede cambiar su contraseña
Prioridad	Baja
Precondición	El usuario debe estar dentro del sistema
Entrada	El sistema pide: Contraseña Actual Contraseña nueva
Salida	Confirmación de cambio de contraseña
Procedimiento	En la sección de usuario, el sistema tendrá la opción de cambio de contraseña.
Postcondición	Contraseña de usuario actual de modifica

Tabla 09. Requerimiento cambio de contraseña.

Fuente: Los autores

Requerimiento	Crear Usuario
Descripción	El sistema debe permitir la creación de nuevos usuarios. Los usuario que cuenten con esta función, son los usuarios con privilegio de administrador.
Prioridad	Alta
Precondición	El usuario debe estar dentro del sistema y con usuario que tenga privilegios de administrador
Entrada	<ul style="list-style-type: none"> • Usuario • Contraseña • Dirección de Correo • Celular • Nombre Completo de Usuario • Tipo de usuario
Salida	Mensaje de Creación exitosa
Procedimiento	El sistema muestra un formulario para nuevos registros con los datos de entrada. Después de ingresar los datos en el formulario, se valida los caracteres y se registra un nuevo usuario
Postcondición	El usuario queda registrado en el sistema y puede iniciar sesión.

Tabla 10. Requerimiento crear usuario.

Fuente: Los autores

Requerimiento	Eliminar Usuario
Descripción	El sistema debe permitir la Eliminación de usuarios. Los usuario que cuenten con esta función, son los usuarios con privilegio de administrador.
Prioridad	Alta
Precondición	Los usuarios que tienen esta función son solo los usuarios con privilegio de administrador
Entrada	<ul style="list-style-type: none"> Nombre de usuario
Salida	Mensaje: Eliminación de usuario exitosa
Procedimiento	El sistema muestra una lista de usuarios registrados, el cual cuenta con un botón para eliminar usuario.
Postcondición	El usuario queda eliminado en el sistema y no puede iniciar sesión.

Tabla 11. Requerimiento eliminar usuario.

Fuente: Los autores

Requerimiento	Crear zona
Descripción	El sistema debe permitir la creación de zonas para vincular con dispositivos sensoriales
Prioridad	Alta
Precondición	Ingresar con usuario Administrador, este usuario cuenta con privilegio para la creación de zonas.
Entrada	<ul style="list-style-type: none"> Nombre de Zona ubicación de zona Descripción
Salida	Mensaje: Creación de zona exitosa, y se lista la zona
Procedimiento	El sistema muestra un formulario para el registro de nuevas zonas, donde se deben ingresar los datos de entrada para luego validar y registrar en la base de datos.
Postcondición	Nueva Zona creada

Tabla 12. Requerimiento crear zona.

Fuente: Los autores

Requerimiento	Eliminar zona
Descripción	El sistema debe permitir eliminar zonas y desvincular con dispositivos sensoriales
Prioridad	Alta
Precondición	Ingresar con usuario Administrador, este usuario cuenta con privilegio para la creación de zonas.
Entrada	<ul style="list-style-type: none"> • Nombre de Zona • ubicación de zona • Descripción
Salida	Mensaje: Se eliminó zona, y se lista la zona
Procedimiento	El sistema muestra la opción para eliminar zonas
Postcondición	Eliminar Zona

Tabla 13. Requerimiento eliminar zona.

Fuente: Los autores

Requerimiento	Editar zona
Descripción	El sistema debe permitir editar/Renombrar zonas
Prioridad	Alta
Precondición	Ingresar con usuario Administrador, este usuario cuenta con privilegio para la editar zonas.
Entrada	<ul style="list-style-type: none"> • Nombre de Zona • ubicación de zona • Descripción
Salida	Mensaje: Se editó zona
Procedimiento	El sistema muestra un formulario para editar zonas, donde se deben ingresar los datos de entrada para luego validar y editar en la base de datos.
Postcondición	Editar Zona creada

Tabla 14. Requerimiento editar zona

Fuente: Los autores

2.5.1.2 Módulo Reportería

Requerimiento	Visualizar reportes de eventos
Descripción	El sistema debe permitir visualizar reportes de eventos generados por los dispositivos en general por zona ubicación.
Prioridad	Alta
Precondición	El usuario debe de tener privilegios de administrador para poder generar un reporte..
Entrada	<ul style="list-style-type: none">• Rango de fecha• zona• ubicación
Salida	El sistema genera reporte de eventos
Procedimiento	El usuario selecciona y filtra la zona y el rango de fecha de cual quiere el reporte de eventos a visualizar.
Postcondición	Se genera reporte de eventos por zonas

Tabla 15. Requerimiento para Visualizar reporte de eventos.

Fuente: Los autores

Requerimiento	Exportar reporte en excel
Descripción	El sistema debe permitir descargar los reportes en formato xls
Prioridad	Media
Precondición	El usuario debe de tener privilegios de administrador para poder generar un reporte..
Entrada	<ul style="list-style-type: none">• Reporte generado
Salida	Archivo .xls
Procedimiento	El usuario genera un reporte y selecciona descargar reporte, el cual se descarga automáticamente en su computador.
Postcondición	Se descarga el archivo formato .xls

Tabla 16. Requerimiento para exportar reporte en formato .xls.

Fuente: Los autores

2.5.2 Requerimientos no funcionales

Id. Requerimiento	Requerimiento	Descripción	Prioridad
RNF-01	Usabilidad	El sistema a desarrollar debe ser web	5
RNF-02	Usabilidad	Deberá soportar el navegador Google Chrome	5
RNF-03	Multiplataforma	El sistema deberá funcionar en diversos dispositivos: Pc's, laptops, tablets, celulares	3
RNF-04	Seguridad	El sistema tendrá un dominio para ingresar.	5
RNF-05	Seguridad	Cada usuario deberá autenticarse para ingresar al sistema, el cual será validado con los registros almacenados en la base de datos	5

Tabla 17. Requerimientos no funcionales.

Fuente: Los autores

2.5.3 Definición de Roles de usuarios del Sistema

2.5.3.1 Administrador: Es aquel usuario que se crea en el sistema, el cual tiene todos los privilegios para realizar configuraciones. Puede crear y eliminar usuarios, crear nuevas zonas, eliminar y editar cada zona.

2.5.3.2 Supervisor: Es aquel usuario que solo puede visualizar el dashboard o módulo de eventos, este usuario no tiene permisos para modificar o configurar.

2.5.4 Casos de Uso

2.5.4.1 Autenticación de usuario

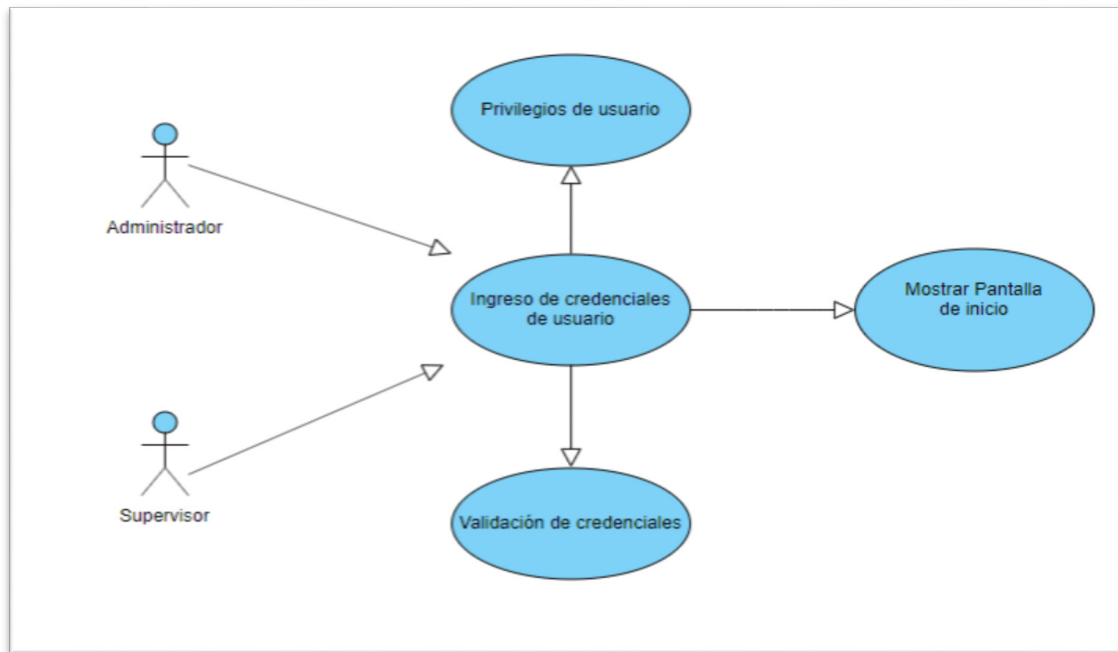


Figura 18: Diagrama de caso de uso de la autenticación de usuario

Fuente: Los autores.

2.5.4.2 Inicio y cierre de sesión

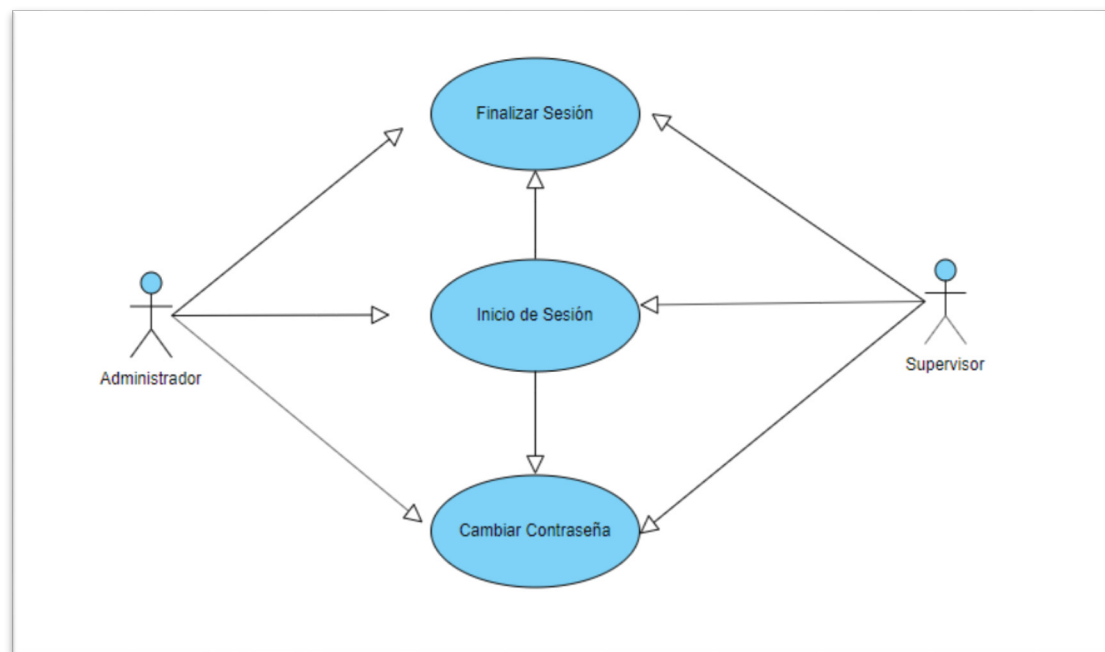


Figura 19: Diagrama de caso de uso de inicio y cierre de sesión.

Fuente: Los autores.

2.6 Procedimiento de armado de maqueta

2.6.1 Preparación de sensores y del circuito Arduino.

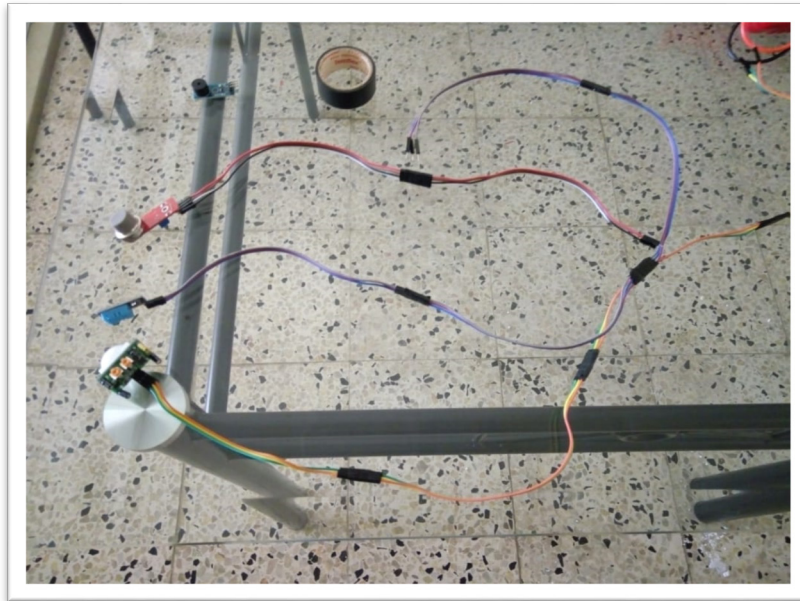


Figura 20: Preparación del cableado de los sensores.

Fuente: Los autores.

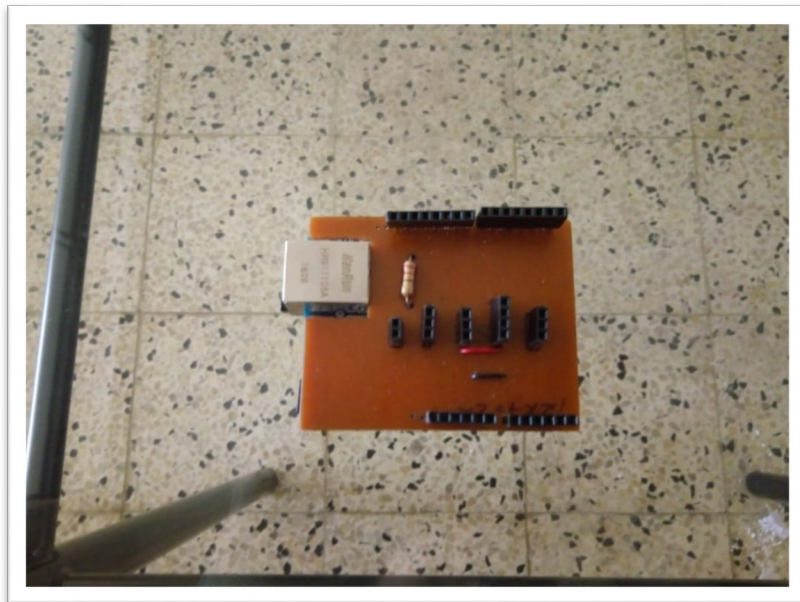


Figura 21: Circuito impreso donde se conectarán los sensores.

Fuente: Los autores.

En la Figura 21. Se observa el circuito impreso en una placa de cobre de fibra de vidrio, aquí es donde se conectarán los sensores.

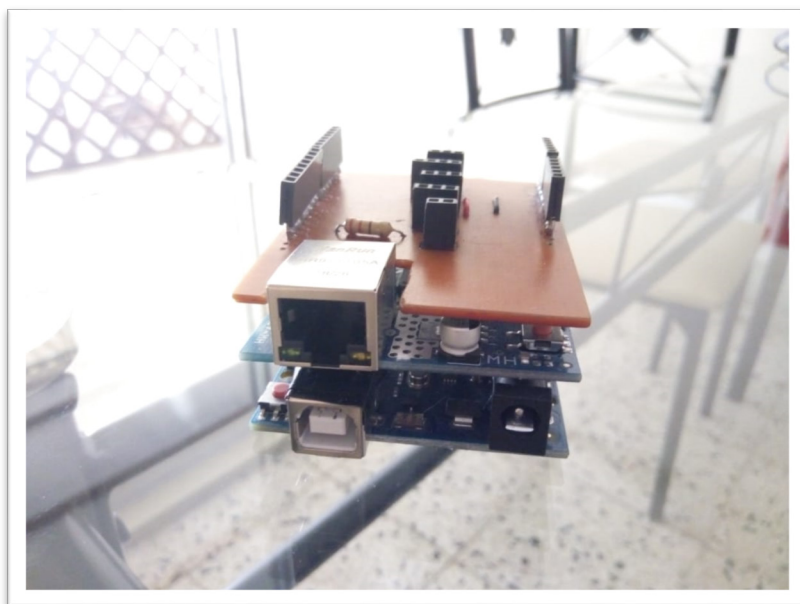


Figura 22: Circuito impreso acoplado como un shield de Arduino.

Fuente: Los autores.

En la Figura 22. Se observan conectados el Arduino uno, el shield ethernet con el cual tendrá conexión con internet y nuestro circuito impreso.

2.6.2 Montaje de los sensores en la maqueta



Figura 23: Maqueta que contendrá el circuito.

Fuente: Los autores.

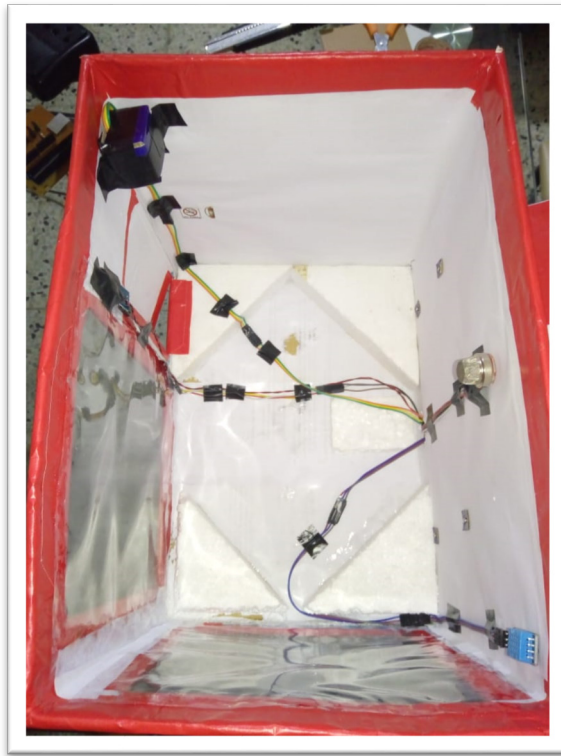


Figura 24: Montado de los sensores en la maqueta.

Fuente: Los autores.



Figura 25: Maqueta correspondiente al primer circuito Arduino (Sede principal).

Fuente: Los autores.



Figura 26: Maqueta correspondiente al segundo circuito Arduino (Sede secundaria).

Fuente: Los autores.

En la figura 25 y 26. Se observan dos maquetas, la primera representará a una sede principal y la segunda a una sede secundaria, estas simularán estar conectadas en una red logrando así el monitoreo de estas en tiempo real.

CAPÍTULO III

PRUEBAS Y RESULTADOS

PROTOTIPO DE SISTEMA DE MONITOREO DE ALARMAS EN UNA RED DE DATA CENTERS

Se implementó un aplicativo web para el monitoreo de los sensores que permitirán determinar los eventos que suceden dentro la red de data centers. Para poder monitorear ingresamos a la URL: <http://monitoreodatacenter.site>.

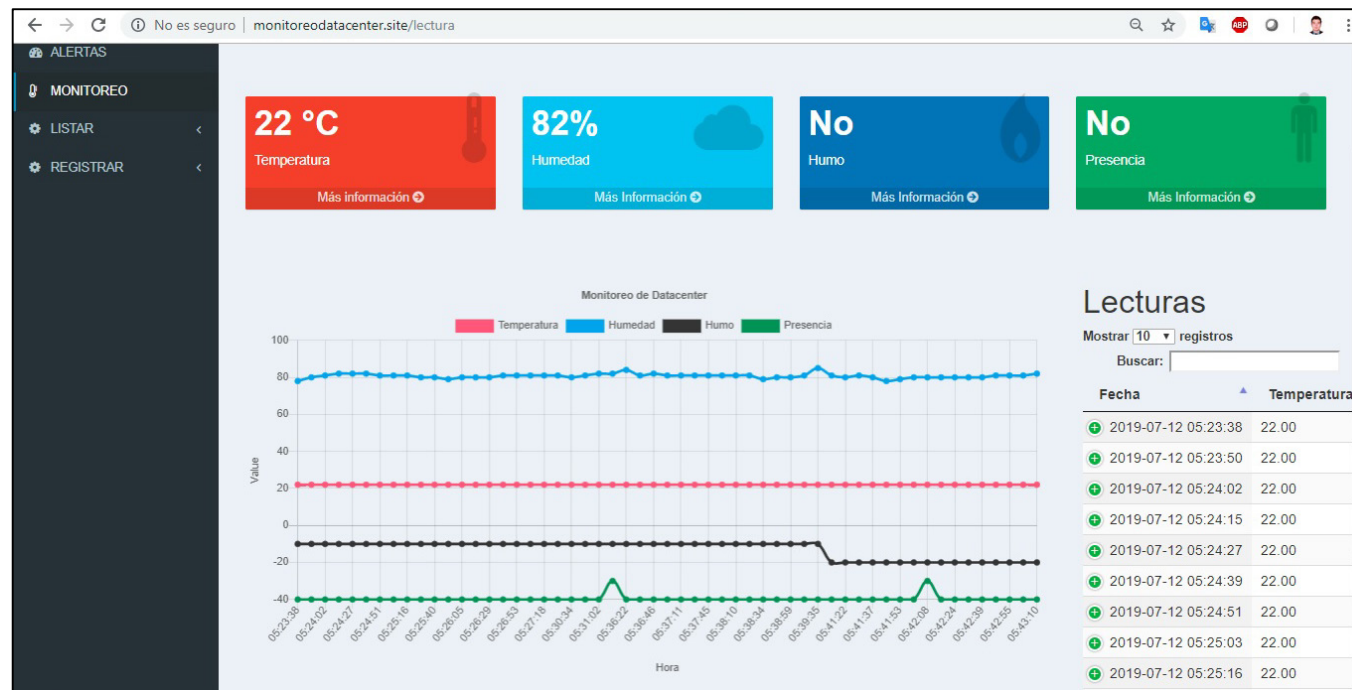


Figura 27: Diseño del Dashboard del sistema de monitoreo de Data Center.

Fuente: Los autores.

3.1 Prueba de verificación de los parámetros leídos por los sensores

Para la verificación de los valores leídos de los sensores se utilizará el monitor serial de propio IDE de arduino, en nuestro caso el arduino lo tenemos conectado al puerto de comunicación COM 8.

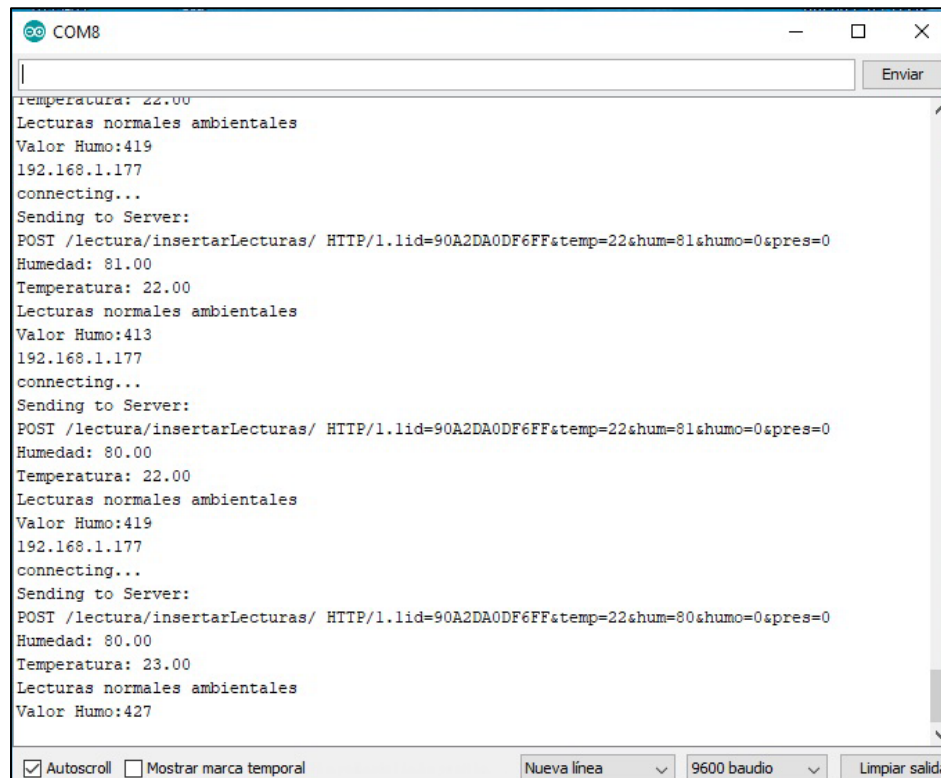


Figura 28: Monitoreo de las lecturas de los sensores mediante el monitor serial.

Fuente: Los autores.

En la Fig. 28 se observa las lecturas obtenidas de cada sensor además de la prueba de conexión al servidor donde se almacenarán.

3.2 Prueba de visualización grafica de los parámetros obtenidos por los sensores.

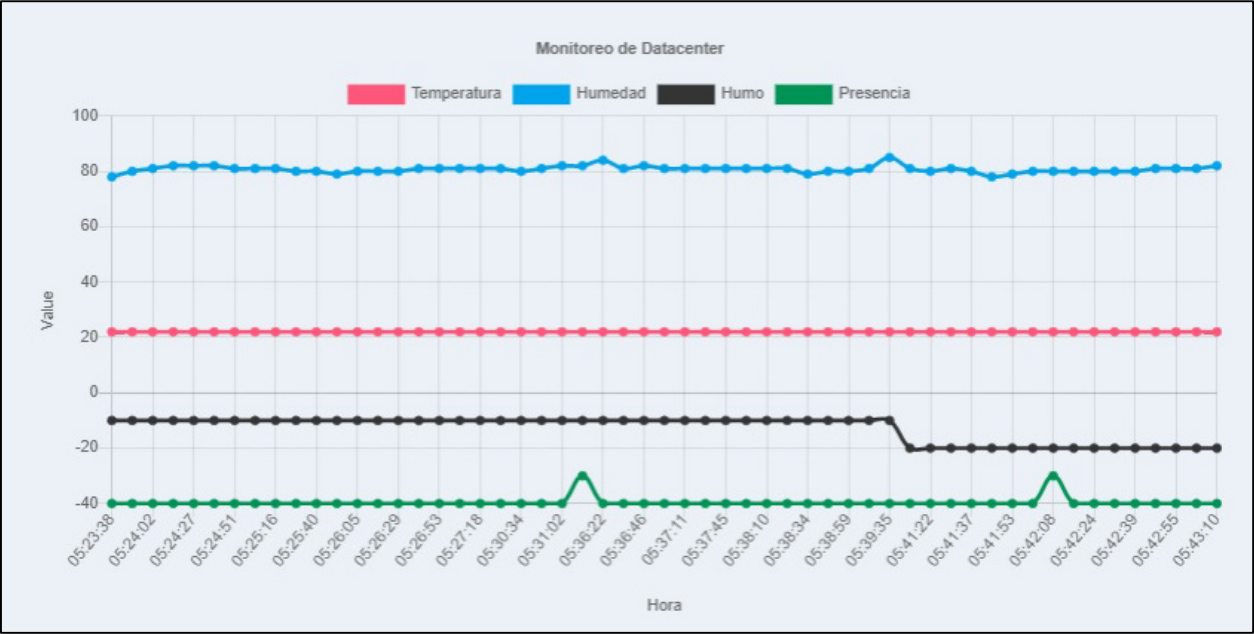


Figura 29: Cuadro de lecturas de los parámetros.

Fuente: Los autores.

En la figura 29. Se observa el cuadro donde se grafican las medidas de los parámetros obtenidos por los sensores de un data center en específico.

Lecturas				
Fecha	Temperatura	Humedad	Humo	Presencia
2019-07-01 18:29:09	16.00	50.00	0.00	0.00
2019-07-01 19:23:04	16.00	50.00	0.00	0.00
2019-07-01 19:23:05	16.00	50.00	0.00	0.00
2019-07-01 19:23:06	16.00	50.00	0.00	0.00
2019-07-01 19:23:07	16.00	50.00	0.00	0.00
2019-07-01 19:23:08	16.00	50.00	0.00	0.00

Figura 30: Cuadro de reporte de los parámetros.

Fuente: Los autores.

En la Figura 30. Se observa el cuadro de reporte de las lecturas de los parámetros en forma de tabla resumen obtenidos por los sensores de un data center en específico.

3.3 Prueba de recepción de correo electrónico.

En la siguiente figura se observa una captura de pantalla de las alertas activas, en este caso se muestra una alerta de Presencia en la Oficina Principal de Sullana la cual será notificada al personal autorizado mediante correo electrónico.



Figura 31: Captura de pantalla de las alertas activas.

Fuente: Los autores

En la figura 32. Se observa la recepción del correo electrónico notificándonos que existe una alerta de humo activa además nos detallará la fecha y hora de inicio de la alarma, el circuito Arduino involucrado, su Mac, el nombre del data center donde ocurrió la incidencia y el nombre de la oficina a la cual pertenece.

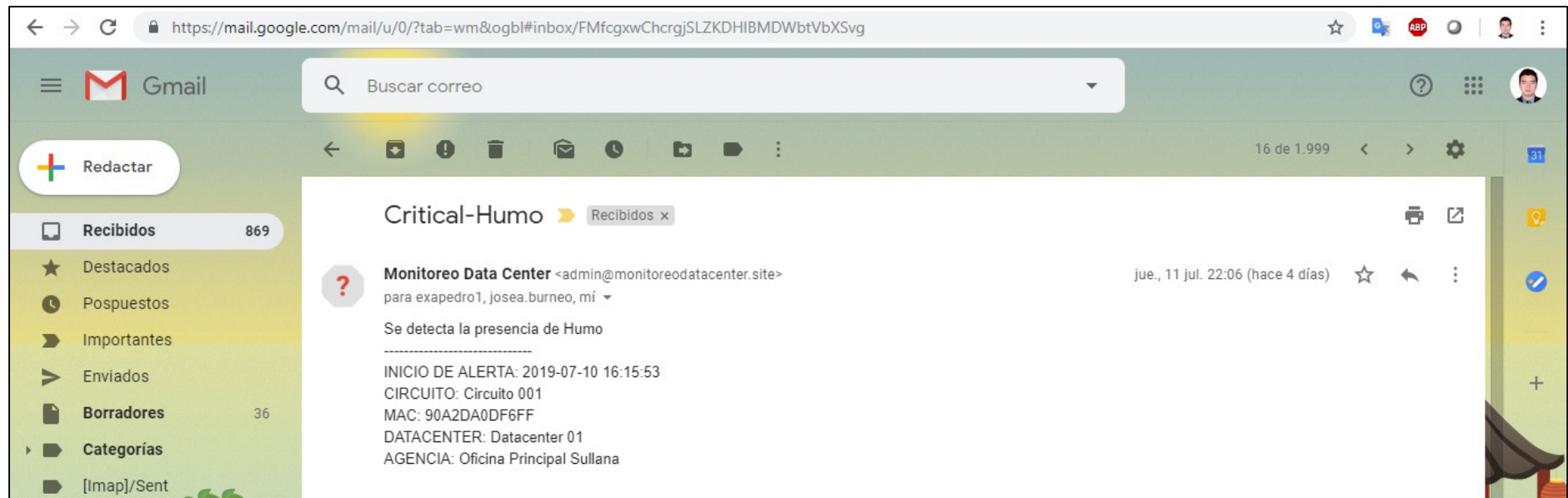


Figura 32: Captura de pantalla de email de alerta de humo.

Fuente: Los autores.

ANEXOS

ANEXO 1: Código para controlar el Arduino uno.

```
#include <Ethernet.h>          //incluimos la librería para hacer uso del shield ethernet.

#include <SPI.h>

#include "DHT.h"               //incluimos librería del sensor de temperatura y humedad.

#define DHTPIN 7               //Definimos el pin digital donde se conecta el sensor

#define DHTTYPE DHT11         //Dependiendo el tipo de sensor

DHT dht(DHTPIN, DHTTYPE); //Inicializamos el sensor de temperatura y humedad

byte mac[] = { 0x90, 0xA2, 0xDA, 0x0D, 0xF6, 0xFF };

byte ip[] = { 192, 168, 1, 177}; //Dirección ip de nuestro primer circuito Arduino.

byte gateway[] = { 192, 168, 1, 1 }; //Puerta de enlace predeterminada.

EthernetClient client;

float temperatura=0;          //inicializamos la variable temperatura.

float humedad=0;              //inicializamos la variable humedad.

boolean presencia=0;          //inicializamos la variable para nuestro sensor de presencia.

boolean humo=0;               //inicializamos la variable temperatura.

int state = LOW;              //por defecto no se detecta movimiento.

const int pinBuzzer = 9; //definimos el pin 9 para que emita el pitido del buzzer.

const int pinled = 12;        //definimos el pin 12 para el led indicativo.

const int sensorPir = 2;      //definimos el pin 2 para que el sensor de presencia .

void setup()

{

    Serial.begin(9600);        //iniciamos la consola serial con velocidad de 9600 baudios.

    dht.begin();               //iniciamos el sensor de temperatura y humedad.
```

```

pinMode(4,INPUT);    // asignamos al pin 4 como modo de entrada de la señal
                    //digital del sensor de humo.

pinMode(pinled,OUTPUT); //asignamos al pinled como modo salida.

}

void loop()

{

leer_parametros(); //iniciamos el procedimiento para obtener las lecturas de los
                    //sensores.

String cadenaMac= "90A2DA0DF6FF"; //almacenamos la dirección mac de
                                   //nuestro primer circuito Arduino en una
                                   //cadena.

enviardata(cadenaMac,temperatura,humedad,humo,presencia);

//enviamos los parámetros.

    if (!client.connected()) //imprime en consola el estado de la conexión a la red.

{

Serial.println();

Serial.println("disconnecting."); //en caso de que no se puede conectar se cierra la
                                   //sesión.

client.stop();

for(;;)

;

} }

void enviardata(String cadenamac,int temp,int hum, int humo,int pres){

delay(5000);

Ethernet.begin(mac, ip, gateway);

Serial.println(Ethernet.localIP());

```



```

delay(1000);

delay(1000);

String dato= "id="+cadenamac+"&";

dato=dato+"temp="+temp+"&";

dato=dato+"hum="+hum+"&";

dato=dato+"humo="+humo+"&";

dato=dato+"pres="+pres;

Serial.println("connecting...");

if (client.connect("93.188.161.158",80)) //Intentamos conectarnos a nuestro
                                     //servidor vps.

{

Serial.println("Sending to Server: ");

client.println("POST /sensor/lectura/insertarLecturas/ HTTP/1.1");

Serial.print("POST /sensor/lectura/insertarLecturas/ HTTP/1.1");

client.println("Host: www.monitoreodatacenter.site");

client.println("Content-Type: application/x-www-form-urlencoded");

client.println("Connection: close");

client.println("User-Agent: Arduino/1.0");

client.print("Content-Length: ");

client.println(dato.length());

client.println();

client.print(dato);

client.println();

}

else

```

```

{
    Serial.println("Cannot connect to Server");
} }

void leer_parametros(){

    humedad=dht.readHumidity();

    temperatura=dht.readTemperature();

    Serial.print("Humedad: ");

    Serial.println(humedad);

    Serial.print("Temperatura: ");

    Serial.println(temperatura);

    ///////////////////////////////////INICIO DE SENSOR DE HUMO////////////////////////////////////

    int adc_MQ = analogRead(A0); //Lemos la salida analógica del MQ

    if(adc_MQ>250){                //el valor de 250 es el límite superior para indicar que
                                   //hasta ese valor no hay presencia de humo.

        Serial.println("Presencia de HUMO en la sala");

        humo=1;

        digitalWrite(pinled, HIGH);

        tone(pinBuzzer,440);

        delay(4000);

        noTone(pinBuzzer);

        delay(500);

    }

    else

    {

        Serial.println("Lecturas normales ambientales");
    }
}

```

```

humo=0;

digitalWrite(pinled, LOW);

}

Serial.print("Valor Humo:");

Serial.println(adc_MQ);

//////////FIN DEL PROCEDIMIENTO DEL SENSOR DE HUMO//////////

//////////NICIO DEL PROCEDIMIENTO PARA LA DETECCIÓN DE PRESENCIA//////////

presencia = digitalRead(sensorPir); //leemos el valor del sensor de presencia.

if (presencia == HIGH) {      // verificamos si está activado el sensor.

    if (state == LOW) {

        presencia=1 ;

        Serial.println("Movimiento detectado!");

        state = HIGH;    // update variable state to HIGH

    }

}

else {

    if (state == HIGH){

        Serial.println("Sin Movimiento!");

        state = LOW;    // update variable state to LOW

    }

}

//////////FIN DE PROCEDIMIENTO PARA LA DETECCIÓN DE PRESENCIA//////////

}

```

ANEXO 2: Diseño del circuito impreso para la placa del Arduino.

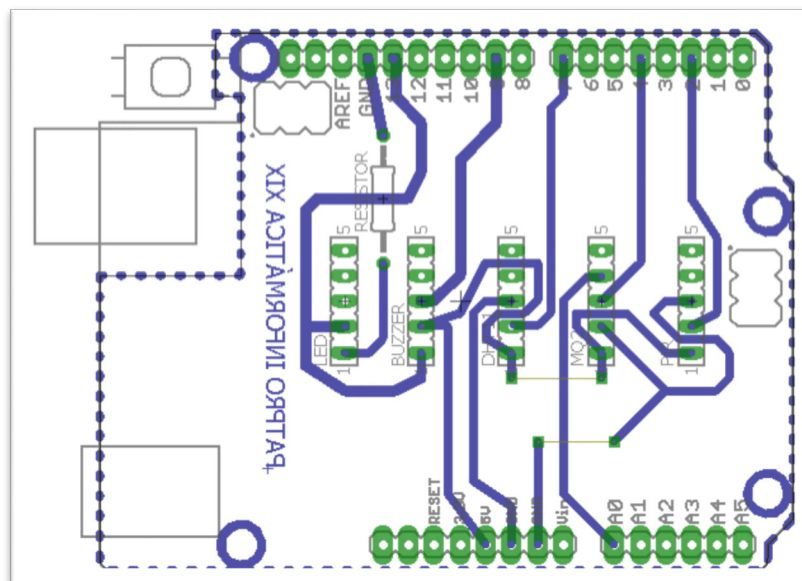


Figura 33: Diagrama del circuito impreso.

Fuente: Los autores.

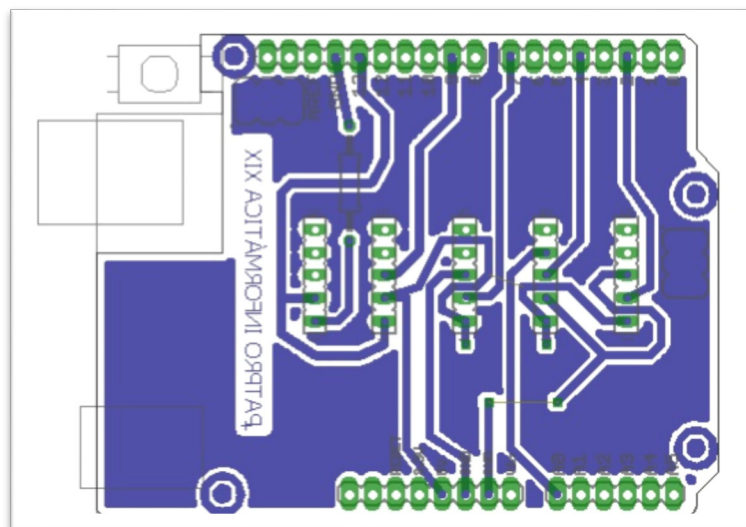


Figura 34: Diagrama del circuito impreso con componentes.

Fuente: Los autores.

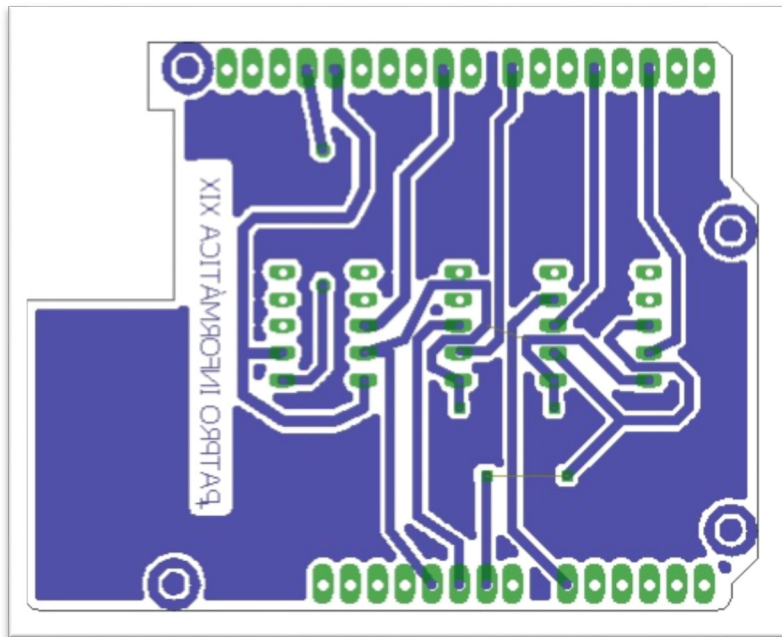


Figura 35: Diagrama del circuito impreso sin componentes.

Fuente: Los autores.

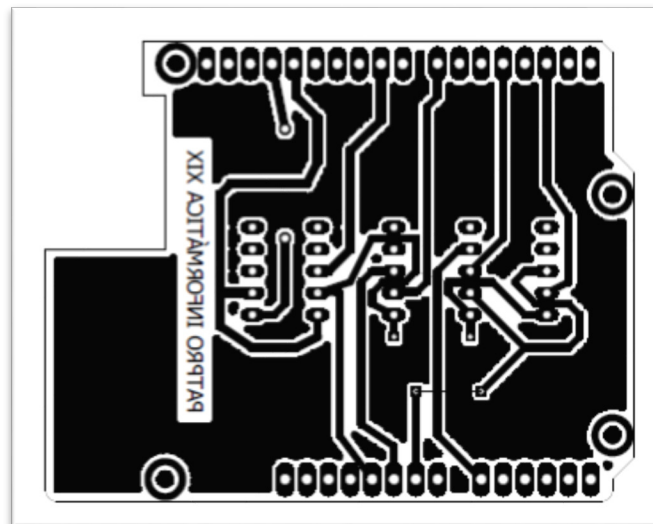


Figura 36: Diagrama esquemático del circuito impreso.

Fuente: Los autores.

ANEXO 3: Modelado de la Base de Datos.

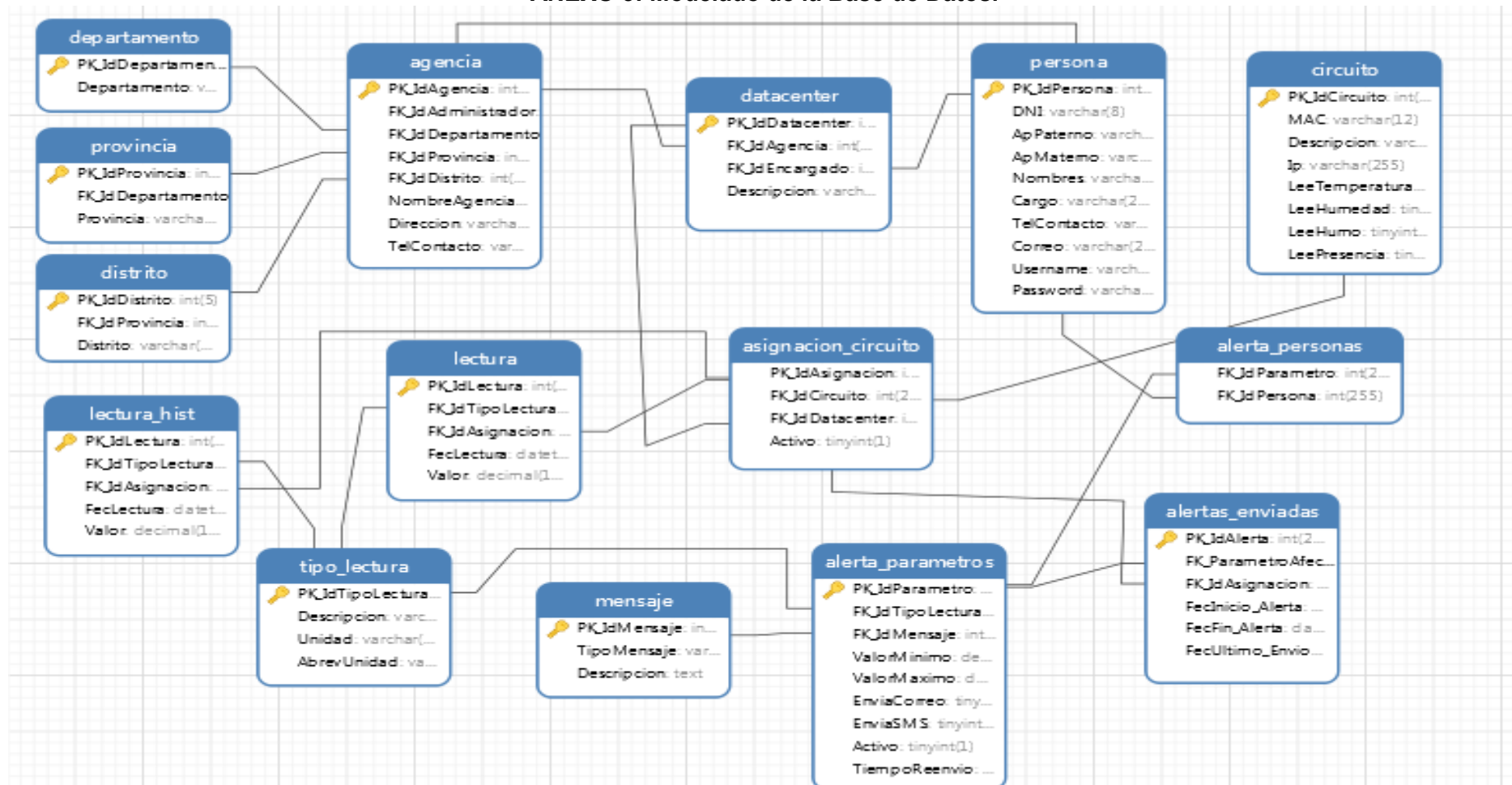


Figura 37: Modelo de la base de datos.

Fuente: Los autores.

ANEXO 4: Captura de pantalla del login del aplicativo web.

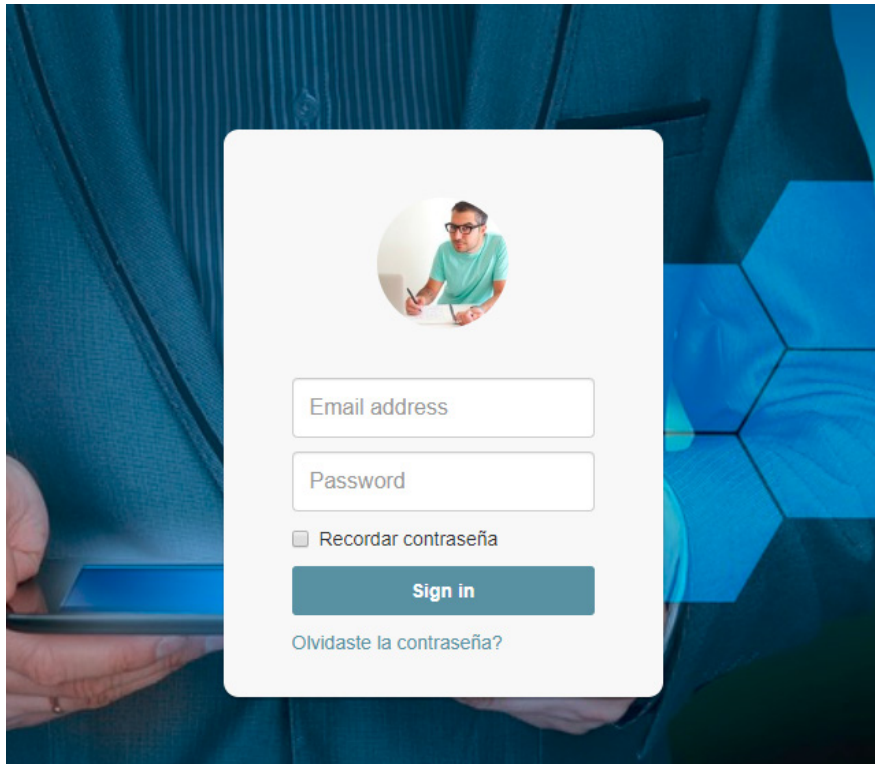


Figura 38: Captura de pantalla del login del aplicativo web del sistema de monitoreo.


Fuente: Los autores.

ANEXO 5: Captura de pantalla del listado de agencias.

← → ↻ ⓘ No es seguro | monitoreodatacenter.site/listAgencia ☆ 🔍 ABP 🔊 🔒 👤

MonitoreoDC

☰



José Burneo
Online

MENU

🔔 ALERTAS

📱 MONITOREO

⚙️ LISTAR ▾

AGENCIAS

USUARIOS

⚙️ REGISTRAR <

AGENCIAS

Mostrar 10 registros

Buscar:

Agencia	Apellidos y Nombres	Ciudad	Telefono
+ Agencia Arequipa	Montes Torres Kenmy	AREQUIPA-AREQUIPA-AREQUIPA	(054) 231716
+ Oficina Especial Bagua	Saavedra Puchulán Pedro David	AMAZONAS-BAGUA-ARAMANGO	123456
+ Oficina Especial HUARAZ	Montes Torres Kenmy	ANCASH-HUARAZ-HUARAZ	123456
+ Oficina Especial Piura	Montes Torres Kenmy	PIURA-PIURA-PIURA	(073)321450
+ Oficina Principal Sullana	Saavedra Puchulán Pedro David	PIURA-SULLANA-SULLANA	(073)284400

Mostrar 1 al 5 , Total: 5 registros

Anterior 1 Siguiente

Figura 39: Captura de pantalla del listado de agencias.

Fuente: Los autores.

ANEXO 6: Captura de pantalla del listado de usuarios.

← → ↻

No es seguro | monitoreodatacenter.site/listUsuario

☆

ABP

MonitoreoDC

☰

10

10

Jose Burneo

Online

MENU

ALERTAS

MONITOREO

LISTAR

AGENCIAS

USUARIOS

REGISTRAR

USUARIOS

Mostrar 10 registros

Buscar:

Apellidos y Nombres	Cargo	Telefono	Correo
Montes Torres Kenmy	Soprtte	1234	11
Saavedra Puchulán Pedro David	Soporte	971734347	exapedro1@gmail.com

Mostrar 1 al 2 , Total: 2 registros

Anterior

1

Siguiente

Figura 40: Captura de pantalla del listado de usuarios.

Fuente: Los autores.

ANEXO 7: Captura de pantalla del registro de agencias.

The screenshot displays a web browser window with the address bar showing 'monitoreodatacenter.site/regAgencia'. The application interface has a blue header with 'MonitoreoDC' and a user profile for 'José Burneo' (Online). A dark sidebar on the left contains a 'MENU' with options: ALERTAS, MONITOREO, LISTAR, REGISTRAR, AGENCIAS, and USUARIOS. The main content area is titled 'CONFIGURACION' and features a 'Crear Agencia' section. Below this title is a sub-header '* Añadir nueva Agencia'. The form consists of several input fields: 'Nombre de Agencia', 'Departamentos' (a dropdown menu), 'Provincias' (a dropdown menu), 'Distritos' (a dropdown menu), 'Direccion', 'Telefono', 'Buscar Usuario', and an empty text field. At the bottom of the form are two buttons: 'Aceptar' (green) and 'Limpiar' (orange).

Figura 41: Captura de pantalla del registro de agencias.

Fuente: Los autores.

ANEXO 8: Captura de pantalla del registro de Usuarios.

The screenshot shows a web browser window with the address bar displaying "No es seguro | monitoreodatacenter.site/regUsuario". The page has a blue header with the "MonitoreoDC" logo and a user profile for "José Burneo" (Online). A left sidebar menu includes "MENU", "ALERTAS", "MONITOREO", "LISTAR", "REGISTRAR", "AGENCIAS", and "USUARIOS". The main content area is titled "CONFIGURACION" and "Registrar Usuario", with a sub-header "* Añadir un nuevo usuario del sistema". The registration form contains input fields for "DNI", "Nombres", "Apellido Paterno", "Apellido Materno", "+51", "correo", and "Cargo". At the bottom of the form are two buttons: "Grabar" (green) and "Limpiar" (orange).

Figura 42: Captura de pantalla del registro de usuarios.

Fuente: Los autores.

ANEXO 9: Diagrama de actividades

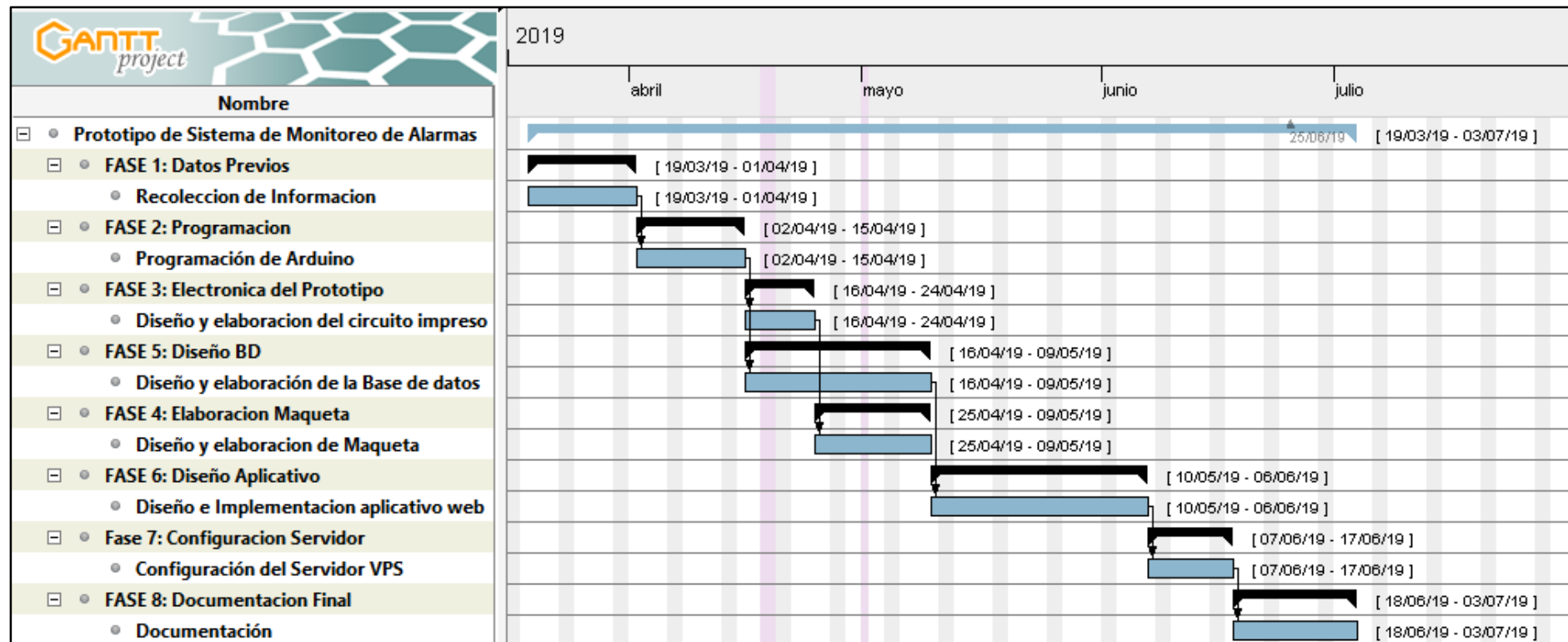


Figura 43: Captura de pantalla del diagrama de Gantt realizado con el programa Gantt Project.

Fuente: Los autores.

CONCLUSIONES

De acuerdo a lo investigado se concluye:

- Es posible diseñar e implementar el sistema electrónico de lectura de parámetros ambientales y de presencia para una red de red de data center.
- Es posible diseñar e implementar el software del prototipo de monitoreo de alarmas y envío de alertas.
- Se pudo comprobar la funcionalidad del prototipo en un ambiente controlado.
- Es económica y técnicamente viable llevar a cabo la ejecución del proyecto debido a la facilidad y utilidad de su diseño, generando beneficios al garantizar un mejor tiempo de respuesta ante la indisponibilidad de los servicios y asegurando la continuidad del negocio.

RECOMENDACIONES

- Realizar periódicamente el cambio de sensores, debido a que con el paso del tiempo y estando expuestos a las condiciones ambientales de la data center, estos pueden empezar a deteriorarse y en consecuencia obtener lecturas anómalas.
- Nuestro escenario se implementó con un servidor VPS el cual nos permite emular un escenario real en producción. Para el despliegue del aplicativo web del sistema de monitoreo dependerá de las políticas de seguridad que tenga cada empresa.
- Para el diseño del prototipo de sistema de monitoreo es fundamental contar con los conocimientos de las especificaciones mencionadas por la asociación de ASHRAE que proporcionan información sobre los parámetros idóneos de temperatura y humedad relativa en la cual debería de estar configurado los data centers.

REFERENCIAS BIBLIOGRÁFICAS

Jadiaz. (21 de enero de 2016). IESCAMP. Recuperado el 19 de mayo de 2019, de <http://www.iescamp.es/miarduino/2016/01/21/placa-arduino-uno/>

Naylamp Mechatronics. (s.f). Recuperado el 19 de mayo de 2019. de <https://naylampmechatronics.com/sensores-temperatura-y-humedad/57-sensor-de-temperatura-y-humedad-relativa-dht11.html>

Jadiaz. (11 de mayo de 2016). IESCAMP. Recuperado el 19 de mayo de 2019, de <http://www.iescamp.es/miarduino/2016/05/11/conectar-arduino-a-una-red-ethernet/>

Nextiafenix. (s.f) Recuperado el 20 de mayo de 2019, de <https://www.nextiafenix.com/producto/sensor-pir-de-movimiento-hc-sr501/>

Nextiafenix. (s.f) Recuperado el 20 de mayo de 2019, de <https://www.nextiafenix.com/producto/sensor-de-gas-mq-2/>

Llamas, L. (10 de julio de 2016). Luis Llamas. Recuperado el 21 de mayo de 2019, de <https://www.luisllamas.es/reproducir-sonidos-arduino-buzzer-pasivo-altavoz>

ASHRAE. (22 de junio de 2016). Data Center Power Equipment Thermal Guidelines and Best Practices.

Grupo Cofitel (s.f) Recuperado el día 21 de mayo de 2019, de <https://www.c3comunicaciones.es/data-center-el-estandar-tia-942/>

GLOSARIO

Data center: Se denomina Centro de Proceso de Datos o Datacenter a aquella ubicación donde se concentran todos los recursos necesarios para el procesamiento de la información de una organización. (DATACENTER-Una mira por dentro, por Víctor Gabriel Galván, 1ª ed. 2013).

Prototipo: Ejemplar original o primer molde en que se fabrica una figura u otra cosa. (RAE – 2018).

GSM: El sistema global para las comunicaciones móviles (del inglés Global System for Mobile communications, y originariamente del francés groupe spécial mobile) es un sistema estándar, libre de regalías, de telefonía móvil digital.

Un cliente GSM puede conectarse a través de su teléfono con su computador y enviar y recibir mensajes por correo electrónico, faxes, navegar por Internet, acceder con seguridad a la red informática de una compañía (red local/Intranet), así como utilizar otras funciones digitales de transmisión de datos, incluyendo el servicio de mensajes cortos (SMS) o mensajes de text.

ARM: es una arquitectura de 32 bits desarrollada en 1983 por la empresa Acorn Computers Ltd para usarse en computadoras personales que maneja un sistema de instrucciones realmente simple lo que le permite ejecutar tareas con un mínimo consumo de energía.

ASHRAE: La Sociedad Americana de Ingenieros de Calefacción, Refrigeración y Aire Acondicionado Ingenieros (ASHRAE) es una asociación profesional que buscan avanzar en la calefacción, ventilación, aire acondicionado y refrigeración (HVAC & R) el diseño y la construcción de sistemas.

Bulbo seco: Temperatura de bulbo seco o temperatura seca es la medida con un termómetro convencional de mercurio o similar cuyo bulbo se encuentra seco.

HVAC: Equivale a lo que en inglés se llama Heating, Ventilating and Air Conditioning, o por sus siglas HVAC, expresión en la que aparecen tres conceptos separados: ventilación y calefacción por un lado y aire acondicionado.

PWM: La modulación por ancho de pulsos (también conocida como PWM, siglas en inglés de pulse-width modulation) de una señal o fuente de energía es una técnica en la que se modifica el ciclo de trabajo de una señal periódica (una senoidal o una cuadrada, por ejemplo).

ICSP: Programación en el sistema (ISP por las siglas del inglés: In-system programming), también conocido como Programación serie en circuito (ICSP por las siglas del inglés : In-Circuit Serial Programming), es la habilidad de algunos dispositivos lógicos programables, microcontroladores y otros circuitos electrónicos, de ser programados mientras están instalados en un sistema completo, en lugar de requerir que el chip sea programado antes de ser instalado dentro del sistema.

SRAM: Son las siglas de la voz inglesa Static Random Access Memory, que significa memoria estática de acceso aleatorio (o RAM estática), para denominar a un tipo de tecnología de memoria RAM basada en semiconductores, capaz de mantener los datos.

EEPROM: Son las siglas de Electrically Erasable Programmable Read-Only Memory (ROM programable y borrrable eléctricamente). Es un tipo de memoria ROM que puede ser programada, borrada y reprogramada eléctricamente

Shield Ethernet: Ethernet Shield permite a una placa Arduino conectarse a internet. Está basada en el chip ethernet Wiznet W5100. El Wiznet W5100 provee de una pila de red IP capaz de TCP y UDP. Soporta hasta cuatro conexiones de sockets simultáneas.

Router: Se trata de un producto de hardware que permite interconectar computadoras que funcionan en el marco de una red. Su función: se encarga de establecer la ruta que destinará a cada paquete de datos dentro de una red informática.

Switch: Un switch es un dispositivo que sirve para conectar varios elementos dentro de una red.

Raspberry: es un ordenador de placa reducida, ordenador de placa única u ordenador de placa simple (SBC) de bajo coste desarrollado en el Reino Unido.

Piroeléctrico: es la propiedad que presentan ciertos materiales que sometidos a cambios de temperatura experimentan cambios en la polarización eléctrica, por lo que dichos cambios de temperatura inducen un campo eléctrico en el interior del material, causado por movimiento de cargas positivas y negativas en los extremos opuestos de la superficie. Esto tiene numerosas aplicaciones prácticas, como por ejemplo la construcción de termómetros electrónicos.

GND: Es la toma de tierra, 0V. AREF (Analog Reference) es el pin que nos suministra la tensión para el rango máximo de los puertos analógicos, normalmente 5V. Vin es la entrada de alimentación de la placa Arduino.

Transistores BJT: El transistor de unión bipolar (del inglés bipolar junction transistor, o sus siglas BJT) es un dispositivo electrónico de estado sólido consistente en dos uniones

PN muy cercanas entre sí, que permite aumentar la corriente y disminuir el voltaje, además de controlar el paso de la corriente a través de sus terminales.

Patch cord: Patch Cord o Cable de Conexión se le llama al cable (UTP, F.O., etc) que se usa en una red para conectar un dispositivo electrónico con otro.

VPS: Un servidor virtual privado (VPS, por sus siglas en inglés) es una partición virtual dentro de un servidor físico. Este método le permite a cada partición individual el uso exclusivo los recursos que le hayan sido asignados.

Dominio: Extensión o dominio de Internet es un nombre único que identifica a un sitio web en Internet.

Dashboard: Es una gráfica que enseña los indicadores fundamentales implicados a la hora de conseguir una serie de objetivos determinados en un negocio.